AD-A091 943 SYSTEMS AND APPLIED SCIENCES CORP RIVERDALE MD F/G 9/2

A SURVEY OF COMPUTER SIMULATIONS OF DIGITAL AVIONICS SYSTEMS.(U)

SEP 80 J WATSON, J DAVIS, R SATTERFIELD F33615-79-C-1870

UNCLASSIFIED AFWAL-TR-80-1057 NL

1 of 1

AD A091 943

END

DATE FILMED

1 8h

DTIC

LEVEL II

AFWAL-TR-80-1057

A SURVEY OF COMPUTER SIMULATIONS OF DIGITAL AVIONICS SYSTEMS

J. Watson
J. Davis
R. Satterfield

Systems and Applied Sciences Corporation
6811 Kenilworth Avenue
Riverdale, Maryland 20840

September 8, 1980

TECHNICAL REPORT AFWAL-TR-80-1057
Final Report for Period July 197 - February 1980

Approved for Release: Distribution Unlimited

Avionics Laboratory
Air Force Wright Aeronautical Laboratories
Air Force Systems Command
Wright-Patterson Air Force Base
Ohio 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Daniel V Ferens

DANIEL V. FERENS
Electronics Engineer
Concepts and Evaluation Group

Terrance A. Brim

TERRANCE A. BRIM
Chief, System Integration Branch
Avionics Laboratory

FOR THE COMMANDER

Raymond E. Siferd

RAYMOND E. SIFERD, Colonel, USAF
Chief, System Avionics Division
Avionics Laboratory

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/AAAS*, W-PAFB, OH 45433 to help us maintain a current mailing list".*

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER AFWAL-TR-80-1057	2. GOVT ACCESSION NO. AD-A094 943
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) A Survey of Computer Simulations of Digital Avionics Systems	5. TYPE OF REPORT & PERIOD COVERED Final Report for Period July 1, 1979 - Feb. 1980
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James Watson, J. Davis, R. Satterfield	8. CONTRACT OR GRANT NUMBER(s) F33615-79-C-1870, P00001
9. PERFORMING ORGANIZATION NAME AND ADDRESS Systems & Applied Sciences Corporation, 6811 Kenilworth Avenue, Riverdale, Maryland 20840	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS P.E.: 62204F Project, Task, Work Unit 2003 09, 15
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Wright Aeronautical Laboratories, Air Force Systems Command, Wright-Patterson AFB, OH 45433	12. REPORT DATE September 1980
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Final rept. 1 Jul 79-1 Feb 80	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Simulation	Simulation Survey
Avionics	GASP
Digital Simulation	SIMSCRIPT
Avionic Simulation	Distributed Processing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Three simulation systems (GCSS, DSDS, and DAS/DDPM) can meet AFWAL's requirements for flexible, all-software, engineering tool. All factors considered, GCSS can meet those needs best. for the short-term, and in so doing, serve as a test vehicle for better definition of AFAL's long-term requirements.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

393826

FOREWORD

This technical report presents the results of a Survey of Computer Simulation of Digital Avionic Systems. The survey was conducted under Contract No. F33615-79-C-1870, P00001, by Systems and Applied Sciences Corporation. This effort was conducted as ~~SASC Task No.~~ 6252-6253.

The work reported herein was performed during the period of 1 July 1979 to 1 February 1980. SASC wishes to thank Mr. Carl Mattes, Naval Air Defense Center, Mr. Robert Jones, Hughes Aircraft Company, Mr. John Piner, National Aeronautical and Space Administration, and Mr. Norman Geer, General Electric Corporation, for their cooperation in support of the survey.

The survey participants and contributors to this report are Mr. J. Davis, Mr. R. Satterfield, and Mr. J. Watson.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

AFWAL/AAAS	Air Force Wright Aeronautical Laboratory/Avionic System Engineering Branch
APL	A Programming Language
ARES	Automated Requirements Engineering System
ASD	Aeronautical Systems Division
AUI	Analysis User Interface
C^3	Command Control Communications
CADSAT	Computer-Aided Design and Specification Analysis Technique
CM	Cost Model
CPU	Central Processing Unit
DAS/DDPM	Design Analysis System/Distributed Data Processing Model
DDC	Defense Documentation Center
DIO	Direct Input Output
DIOD	Direct Input Output Device
DMA	Direct Memory Access
DSDS	Data System Dynamic Simulator
DSEM	Data System Element Model
DUI	Design User Interface
DUM	Device Utilization Model
ESD	Electronic Systems Division
G BUS	General Bus
GCSS	General Computer System Simulator
GEC	General Electric Corporation
HAC	Hughes Aircraft Corporation
IM	Integrated Model
IPP	Input Pre-Processor
LDS	Lockheed Dialog Service
MD or MEM	Memory Device
MM	Memory Model
MSFC	Marshal Space Flight Center
MUX	Multiplex

NADC	Naval Air Development Center
NASA	National Aeronautical and Space Administration
OM	Operations Model
PARCST	DSDS Cost Model Parameters
PARDUM	DSDS Device Utilization Model Parameters
PAROM	DSDS Operations Model Parameters
PARTRM	DSDS Throughput Resource Utilization Model Parameters
PE	Processing Element or CPU
PERCAM	Performance and Configuration Analysis Methodology
PIO	Processor Input/Output
PIOB	Processor Input/Output Bus
PIOD	Processor Controlled Input/Output Device
PLS	Product Line Simulation
RADC	Rome Air Development Center
TIM	Technical Interchange Meeting
TRUM	Throughput Resource Utilization Model
UIV	User Initialized Variable

SECTION I

SURVEY OF COMPUTER SIMULATIONS OF DIGITAL AVIONIC SYSTEMS

1.0 INTRODUCTION

This report describes the activities and results of tasks performed by Systems and Applied Sciences Corporation in conducting a "Survey of Computer Simulations of Digital Avionic Systems". The report is organized into six sections. The remainder of this section provides background information regarding AFWAL/AAAS motives for initiating the survey. The survey objectives and scope are discussed in Section II. Section III addresses the survey strategy and approach and details criteria used to screen candidate simulation programs. In Section IV, survivors of the screening are presented with detailed critiques regarding potential AFWAL/AAAS applicability accompanied by pertinent design and descriptive data. Included are assessments in terms of avionics applicability and accessibility to AFWAL/AAAS. Section V presents recommendations based upon the survey results. Section VI is a compendium of simulations evaluated during the survey. A bibliography identifying pertinent citations and articles reviewed during the survey is also included.

1.1 BACKGROUND

Within the simulation community, a variety of simulation modeling tools have been developed to support the investigation of systems and parameters that have design and processing characteristics similar to avionic systems. Their applications include distributed processing, information transfer, and fault tolerance. Unlike modeling tools of the past, including some currently being used, these systems have been built with general purpose applications and user requirements in mind. These models are user-oriented

in that they are designed with libraries of sub-models and utilize methods of system definition intercommunication and interface techniques that reduce time-consuming set-up and run times. In short, these models encourage their use by design engineers.

SECTION II

SURVEY GROUNDRULES

2.0 OBJECTIVES

The Survey of Computer Simulations of Digital Avionic Systems had two objectives. The primary objective was to identify state-of-the-art simulation techniques and models with digital avionics simulation applications. The resultant models must exhibit capability to effectively support analysis in avionic network design, performance or cost. They must further possess design features and characteristics that enhance user interface and model/problem definition.

The secondary objective was to provide a compendium of recent works in digital simulation that involve network designs, distributed processing, etc. The compendium would provide AFWAL with an awareness of current efforts and centers of expertise within the simulation community; developers and users.

2.1 SCOPE

The scope of the survey established by the contract Statement of Work required SASC to consider a variety of simulations and models independent of application and source.

Simulation is being used as an analysis tool by government, industry, and academic organizations for a variety of applications. This fact, combined with the number of meetings, journals, and societies concerned with the various aspects of simulation, represents a large source of candidate simulations.

Direct utility/use of a simulation to AFWAL/AAAS was an important aspect of the survey. Therefore, constraints were imposed upon the selection of candidate simulations. Of particular interest to AFWAL/AAAS were simulations that are:

1) All software: All modeling and simulation accomplished by software. Hardware limited to user interface and peripherals.
2) Compatible with available computational resources: i.e., DEC-10, CDC 6600 computer, and GASP IV, SIMSCRIPT, or FORTRAN software.
3) Applicable to avionics analysis in terms of investigating the aspects of distributed processing, information transfers and/or fault tolerant systems.
4) Operational in terms of status and validity.

To satisfy the primary goal of the survey (identification of design tools utilizable by AFWAL/AAAS) the above constraints were viewed as guidelines rather than mandates. Therefore, consideration was given to a variety of simulations that did not necessarily satisfy all of the above criteria.

SECTION III

SURVEY APPROACH

3.0 TASKS

This section addresses the approach and methods used in performance of the tasks. It includes discussions of three principal tasks that comprised the effort. In addition, pertinent output data, along with screening criteria used to assess the various candidates, is identified. The discussions of the survey tasks are presented in the order of execution. They are:

1. Identification of Data Sources
2. Data Acquisition
3. Data Analysis

3.1 IDENTIFICATION OF DATA SOURCES

SASC's first task was to develop an information data base that included preliminary identification of candidate simulations and models and potential sources. This data base took the form of a preliminary Master List consisting of candidate simulations that grossly exhibited potential avionics and analysis applicability. This preliminary Master List was the basis for later activities, and was modified over the course of the study to include only the best information available. The Bibliography of this report contains the Master List in its final form. Two methods were used to identify the candidates. They involved:

1. Automated data base searches
2. Open literature searches

Automated data base searches were conducted at Wright-Patterson Air Force Base using the Defense Documentation Center (DDC), and Lockheed Dialog Service (LDS) data base retrieval systems. A strategy was derived by developing key word descriptors

to facilitate each search. The descriptors were derived in part from the requirements specified in the Statement of Work, and from items and areas of interest pertinent to the objective of the survey. The descriptors used included:

Avionics	Information Transfer
Simulation	Digital
Networks	Computer
Time Division Multiplex	Fault Tolerance
Systems	SIMSCRIPT
Models	GASP
Distributed	Q-GERT
Architecture	Processors

Several factors required that the search process be performed repetitively. This was necessary because of: a) the number of possible combinations of search terms, b) the number and types of topics being sought, and c) the possibility that within a data base, similar topics could conceivably be referenced using different, yet similar, terminologies (for example, the terms computer and processor, or simulation and model are oftentimes used synonymously). Indeed, this was experienced on several occasions. Therefore, to be as effective as possible, it was necessary to iterate using variations on the descriptors.

The open literature search was conducted concurrently with the DDC and LDS searches. Data was located at the Air Force Institute of Technology Library, the Air Force Wright Aeronautical Laboratory library, and the Wright State University and University of Dayton libraries. The approach employed was based upon scanning pertinent simulation literature journals, scientific proceedings, etc., and to record pertinent facts relevant to the survey objectives for future reference.

Documents reviewed during the open literature search were from a variety of government and scientific community sources. Government sources included the National Technical Information

Service (NTIS) and the Scientific and Technical Abstract Reports (STAR). Scientific sources were both academic and industrial: various transactions of the IEEE, various publications of ACM Special Interest Groups (such as Simuletter), non-affiliated journals such as Computer Networks, and the proceedings of engineering and data processing conferences and symposia.

Since the eligibility of the candidates was based principally on the inferences of titles and abstracts, further screening was required. To facilitate the screening process, a set of screening criteria was derived and descriptive data regarding each candidate acquired. These items are discussed in the following section (Acquisition of Data).

3.2 ACQUISITION OF DATA

This activity located and abstracted the citations referenced in the Master List described in Section 3.1. It combined aspects of the previous activity and the following analysis task. Articles referenced in the Master List were located and acquired. Location and acquisition was accomplished in several ways. Data was ordered from DDC, and articles were copied from journals and other pertinent literature. Authors of articles were contacted and data subsequently received through the mail. During the reviews, references made to other citations and sources found in the principal documents were also located and reviewed. Further, in discussions with some sources, identification of other potentially applicable simulation developments and programs were brought to the attention of SASC. However, the survey revealed no simulation that singularly met all of the technical criteria described in Section 3.3.

3.3 ANALYSIS

The analysis task resulted in identification of simulation programs that most appropriately exhibit AFWAL/AAAS applicability. These are identified in Section IV. The selection process was based upon the following criteria, which encompass the technical objectives outlined in the Scope (Section 2.1).

1) All Software - This was a firm requirement specified in the Statement of Work. Therefore, to receive mention in this section, the simulation program had to be all-software. Simulation efforts utilizing special hardware/software approaches did not receive consideration.

2) Applicability - This refers to the simulation's ability to effectively support the investigation of data processing, distributed processing, time division multiplex or fault tolerant systems.

3) Availability - This refers to the simulation's design status: whether it is fully or partially operational or in development. Also, accessibility is considered in terms of: compatibility with existing computational hardware and software resources, or whether AFWAL/AAAS access is attainable through other means such as terminals or other available communication nets.

The simulations that were found to be minimally acceptable in accordance with above requirements are described in the Compendium (Section VI) in terms of their inherent attributes and the relevance of these attributes to AFWAL/AAAS desired applications. The attributes reflect the technical and design aspects a user requires to effectively model and simulate a particular problem.

Data used in making these determinations included program descriptions, specialized reports, user manuals, and in some instances, marketing literature. In short, the maximum that could be obtained from the respective source. There were instances

where SASC was only able to obtain a single document describing a particular simulation; such as a program listing or user's manual. For these conditions attempts were made to: (a) obtain more data, and (b) arrange for a technical discussion with the respective source. Thus the discussion of simulation attributes is based upon documented and undocumented factors.

Simulation Attributes

Five attributes or factors were identified as the most relevant for describing and evaluating the candidate simulations. A description of each attribute follows:

1) Performance - Refers to computer time required to compile and execute a problem, as well as the ratio of CPU time to simulated time. This ratio varied widely among the systems surveyed: the three systems discussed in the following section had a range of ratios from 15:1 to 2:7. No absolute standard of performance was utilized, due to the difficulty in even locating simulations that met the criteria described earlier in this section. In the ensuing discussions, performance will be addressed with regard to AFWAL applications and, where possible, correlated to the potential avionics application.

2) Simulation Capacity - Denotes the ability of the simulation to accommodate large models. For example, for a distributed network, are there unreasonable limitations placed on the number of nodes that can be simulated because of restrictive design or host system memory requirements?

3) Ease of Use - Refers to user/simulation interface requirements. Factors considered here include:

   a) Simulation Set-Up Time. This considers the overall effort and time required to learn the system, and to subsequently build and develop a model from a design.

b) Input Language Requirements. This considers how readable the input language is and how quickly models/problems can be defined for input.

c) Diagnostics. This refers to the ability of the simulation to recognize and effectively flag and report input errors of format and content.

d) Flexibility. This considers the design aspects of the simulation as it relates to providing libraries of routines from which to develop models for different applications, and providing the user options to select different levels of simulation (i.e., functional level or detailed interpretive level).

e) Output and Reports. This relates to the usability and range of output data provided by the simulation. Considered here are readability and formats, ability to provide output via charts, graphs, or computerized graphics.

4) Accessibility - Addresses the requirements for hosting the simulation on AFWAL/AAAS available resources and the requirements for obtaining access to the simulations not compatible with these resources.

5) Documentation - Refers to the availability of information that describes the simulation in terms of user interface requirements and capabilities, and system design descriptions and maintenance manuals.

SECTION IV

SURVEY RESULTS

4.0 CANDIDATE SIMULATION/MODELING SYSTEMS

The survey has resulted in the identification of three (3) simulation and modeling systems that exhibit the technical and operational potential to satisfy AFWAL/AAAS analysis requirements. Each candidate is all-software, accessible to the Air Force, and operational in terms of past, current, and intended future use in supporting design analysis activities. The three systems are:

1) Generalized Computer System Simulator (GCSS)
2) Data System Dynamic Simulator (DSDS)
3) Design Analysis System/Distributed Data Processing Model (DAS/DDPM)

Special Note: The DSDS and DAS/DDPM were found as a result of an activity being conducted by ASD/MITRE, Hanscomb Air Force Base, Massachusetts. ESD/MITRE is conducting a program consisting of the identification, evaluation and acquisition of a computer communications system performance model to effect conceptual phase analysis of C^3 systems. The identification and evaluation efforts have been completed.

Two additional systems were included in the ESD/MITRE effort: Performance on Cost Analysis Model (PERCAM) and Automated Requirements Engineering System (ARES). These systems were eventually dropped by this study because of their current operational status. The specific reasons are provided in Section VI (The Compendium).

4.1 GENERALIZED COMPUTER SYSTEM SIMULATOR (GCSS)

4.1.1 INTRODUCTION TO GCSS

The GCSS represents the only "find" that was designed specifically with avionic networks and processing as the target

function. The system was identified during the DDC search.

The GCSS is operational under the auspices of the Naval Air Development Center (NADC) Warminster, Pa. The system will be used to evaluate alternative system architectures and signal processing requirements for the V/STOL avionics system.

The system is a product of Honeywell's Aerospace and Ground Systems Group, Minneapolis, Minnesota. During the past year it has been undergoing enhancements and modifications. These include implementation of an interactive system, improving the user interface and mods to make it work as intended.

The system is accessible to the AFWAL, via the ARPANET. The system's software support requirements preclude hosting on WPAFB computers. At NADC the GCSS is hosted in a CDC 6600. However, it is programmed in SNOBOL and SIMSCRIPT I.5; neither language is supported locally. Therefore, local hosting is not feasible. On the other hand, both WPAFB and NADC CDC 6600 systems are nodes on the ARPANET. Thus access to the system can be achieved remotely. A detailed discussion of GCSS is presented in the following section.

4.1.2 GCSS DESCRIPTION

<u>Source</u>

Naval Air Development Center, Warminster, Pennsylvania

<u>Contact</u>

Mr. C. Mattes, Mr. W. Garrison

<u>Application</u>

Computer Architecture, busing arrangements signal processing.

<u>Reference Data</u>

NADC "Installation Validation and Support of the Honeywell Generalized Computer System Simulator (GCSS), Final User's Manual Contract No. N62269-77-C-0179: Honeywell Systems and Research Center, Aerospace and Defense Group, April 19, 1979.

Note: This manual addresses the batch version of GCSS. The interactive system is not documented.

Description

GCSS is an all-software simulation system that enables the user to evaluate the throughput and occupancy of distributed processing system models as functions of hardware and software. The system is hosted on a CDC 6600 computer and is programmed in SNOBOL and SIMSCRIPT I.5.

The principal purpose of GCSS is to assist in the evaluation of an alternate computer and system architectures and busing arrangements. That is, the simulator provides a design aid for computer systems' performance by measuring throughput and conflicts as a function of factors that include element interconnections, modularity, instruction repertoire and circuit memory technologies. User documentation states that "GCSS is best suited for use after the system level requirements have been defined and acceptable architectures have been determined, and prior to preparation of hardware specifications".

Three major components and steps comprise and constitute a complete GCSS execution. They are:

- IPP - Initialization Pre-Processor
- SORT/MERGE - External Events Ordering
- GCSS - Simulator

Figure 1 presents a pictorial representation of this environment.

The IPP process formats and derives initialization information from user provided input data. There are two types of input required. First, there are thrity-one (31) User Initialized Variables (UIVs) which describe the physical and operational characteristics of the modeled system. Although many UIVs have complex functions, for example MCHAN which describes all job modules in the simulation, most are straightforward: RSTA is a flag array, MDIO is the number of direct I/O devices in the simulation. Table 1 contains a list of all UIVs, and a brief description of each. Second are the simulation control and events data that are input via the external events portion of the input data. These data are executed on, reformatted, crosschecked for errors, etc. The resultant output serves as the input to SORT/MERGE and GCSS functions.

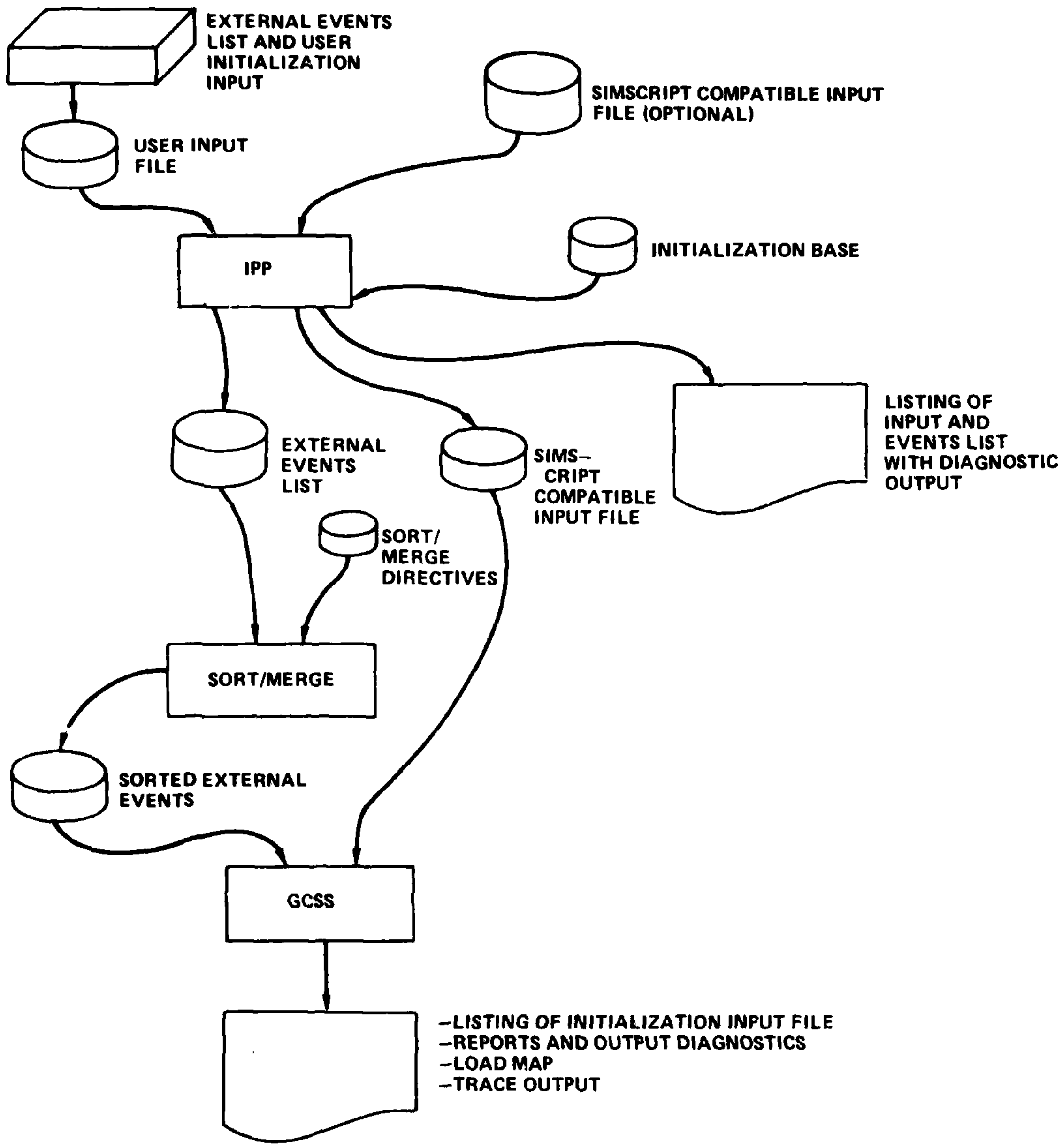


Fig 1 THE GCSS BATCH RUN ENVIRONMENT

TABLE 1. USER INPUT VARIABLES OF GCSS

	NAME	USE
1.	NPE	Number of Processing Elements in the Simulation
2.	NMD	Number of Memory Devices in the Simulation
3.	NPIO	Number of Program I/O Devices in the Simulation
4.	NDIO	Number of Direct I/O Devices in the Simulation
5.	NMODS	Number of Program Modules in the Simulation
6.	MILEV	Maximum Interrupt Level in the System
7.	NBVSS	Number of General Buses in the Simulation
8.	NPIOB	Number of Program I/O Buses in the Simulation
9.	MEMCY	Array of Read and Write Times for Each Memory Device
10.	MCHAN	Central Array of Simulation, Describing all Modules in the Simulation
11.	IOCYC	Array of Cycle Times for Each PIOD
12.	SELMD	Array Ordering Modes (Round Robin, Random, Priority) for Bus Selection
13.	ORSTT	Table Setting ORed Successor Probabilities and Paths
14.	IREPT	Array Describing the Processor Instruction Repertoire
15.	IMIX	Table Describing the Instruction Mix for Each Module
16.	PDUV	Vector Ordering PEs or DIODs for Selection by Scheduler
17.	MDUV	Vector Ordering MDs for Selection by Scheduler
18.	PIOUV	Vector Ordering PIOs for Selection by Scheduler
19.	IRESP	Array Fixing Responding Modules for Each Interrupt Level
20.	BWTIM	Delay Time Before Processors Access (Read or Write) General Bus
21.	PBWTM	Delay Time for Processor Elements Access (Read or Write) General Bus
22.	MISET	Vector Establishing Interrupt Schemes for Each Module
23.	PEBM	Array to Show Interconnections of PEs (Row) to General Buses (Colm)
24.	MDGBM	Array to Show Interconnections of MDs (Row) to General Buses (Colm)
25.	PEPCM	Array to Show Interconnections of PEs (Row) to PIOBs (Colm)
26.	PIPCM	Array to Show Interconnections of PIODs (Row) to PIOBs (Colm)
27.	DIGBM	Array to Show Interconnections of DIODs (Row) to General Bus (Colm)
28.	MAOPT	Logical Switch Noting Fixed or Dynamic Memory Device Assignments
29.	PIOPT	Logical Switch Noting Fixed or Dynamic PIO Device Assignments
30.	BSUL	List Ordering Devices for Bus Utilization
31.	RSTA	Flag Noting Requirement to Output a Particular Report

The SORT/MERGE function orders the external events according to predefined time fields specified with the appropriate event. The result is an ordered external events list according to the time of occurrance associated with each event of the system being modeled. This output serves as an input to GCSS.

GCSS is the simulator. GCSS accepts and executes upon the model characterized by the sorted external events data from SORT/MERGE, and the initialization data output from IPP. The resultant output is a series of reports and traces which allows the user to evaluate the model's performance.

Examples of the types of processing structures that can be evaluated by GCSS are shown in Figures 2.a and 2.b. The user describes the system to be simulated in terms of the number and characteristics of the processing elements (PE's), direct I/O devices (DIO's), memory devices (MEM's), program controlled I/O devices (PIO's), and the interconnections between: a) PE's and PIO's, b) PE's and MEM's, and c) DIO's and MEM's. These elements, which are the basic building blocks of GCSS, are described as follows:

- PE - Processing Elements or CPUs
- DIO or DIOD - Direct I/O Devices; another name for a Direct Memory Access device.
- PIO or PIOD - Program Controlled I/O Device; data transferred under control of a PE. (An anology can be drawn between a PIOD and an avionics sensor.)
- MEM or MD - Memory Device; passive element of a model. Data is exchanged with a memory device either a word at a time or in blocks of user specified size under control of a PE or DIOD.
- PIOB - PIO Bus; buses that connect PEs with each other or with PIODs.
- GBUS - General Bus; buses that connect primarily devices with each other and memory devices (PEs and DIODs.)

Thus distributed architectures, multicomputers, and multiprocessor configurations can be modeled by stipulating types of hardware elements and their respective interconnections.

In addition to the hardware structures, GCSS requires definition of the structure of the applications software. The characteristics of the system's software/program modules and the interrelationships between modules. Modules include instruction mix, total number of instructions, amount of data being passed, interation intervals, and priorities. Module interrelationships are described by specifying the modules that must succeed a given module and the number of predecessors that require completion prior to initiating a module. These and the physical system characteristics are defined via the user provided UIV's and external events data.

GCSS provides the user the option to determine the level of simulation detail desired. These levels can range from interactions on a module to module basis to representation of every memory cycle of every instruction. GCSS allows the user to specify module execution characteristics in terms of a block of execution time down to a string of memory read and write cycles.

Simulation output reports are tabular. They include measures of the resource utilization per program module, percent utilization of processing elements and storage capacity, bus traffic levels, queue sizes and time delays. Specific outputs available at various levels include:

- Statistics on the average amount of time required to complete each module or program section.
- Average number and duration of interrupts that occur throughout the run.
- Scheduling delay statistics.
- Utilization statistics on each piece of hardware.
- Size statistics on the various queues that develop throughout the system.

A sample of GCSS outputs is shown in Table 2.

TABLE 2. SAMPLE OF GCSS OUTPUT REPORT

PROCESSOR ELEMENT UTILIZATION STATISTICS

AT SIMULATED TIME 1.0336573 SECONDS

	By Processor Element No. 1	By Processor Element No. 2	By Processor Element No. 3
Total Number of Memory Requests Made	6982	11500	0
Total Number of Memory Requests Denied	0	0	0
Total Time Spent Waiting for Buse Memories	2033.986 USEC	10176.789 USEC	0.000 USEC
Average Time Spent Per Request	0.291 USEC	0.885 USEC	0.000 USEC
Total Time Spent Waiting for Bus to Memory	0.000 USEC	0.000 USEC	0.000 USEC
Average Time Spent Per Request	0.000 USEC	0.000 USEC	0.000 USEC
Total Number of Program I/O Requests Made	3524	1234	11108
Total Number of Program I/O Requests Denied	0	0	0
Total Time Spent Waiting for Busy I/O Devices	0.000 USEC	0.000 USEC	0.000 USEC
Average Time Spent Per Request	0.000 USEC	0.000 USEC	0.000 USEC
Total Time Spent Waiting for Bus to I/O Devices	1007.201 USEC	472.336 USEC	1604.584 USEC
Average Time Spent Per Request	0.286 USEC	0.380 USEC	0.144 USEC
Composite Processor Utilization Level	1.92 % BUSY 98.08 % IDLE	2.53 % BUSY 97.47 % IDLE	2.07 % BUSY 97.93 % IDLE
Module Currently Being Processed	2	34	16

When modeling any system, GCSS requires initialization of the thirty-one (31) UIV's. These user defined variables are derived from the model's hardware and program module (software) physical and operational characteristics. The hardware characteristics are derived from diagrams of the type shown in Figure 2(a) and 2(b). This allows the user to specify the various types of elements and their interconnections. To specify the program module characteristics, the user must develop a concept of how the system is to operate. Specifically, the user defines the order in which bus selection algorithms are enforced, which priorities are assigned to hardware units to resolve conflicts, and what unique groups of primary devices are required to accomplish each of the jobs to be performed. Additionally, definitions of the number of modules in the system, module "interation rates for specific modules, and module execution sequences are required". Software control flow diagrams called "chaining diagrams" are used for these purposes.

Chaining diagrams depict the model environment by specifying the interdependency of the various program modules. Critical UIV data is derived from the chaining diagram. Each module is uniquely numbered which allows the user to specify unique operational characteristics to each module and assign modules to PE's.

Program chaining is based on four building blocks. These are:

- ANDed Successor
- Statistical Successor
- ANDed Predecessor
- ORed Successor

Examples of these appear in Figure 3. Figure 4 is an example of how these four building blocks can be combined to form a subsystem model.

ANDed Successor means all modules chained to a particular module are enabled when that particular module is completed.

Statistical successor allows conditional branching to a particular module a percentage of time based upon a random number.

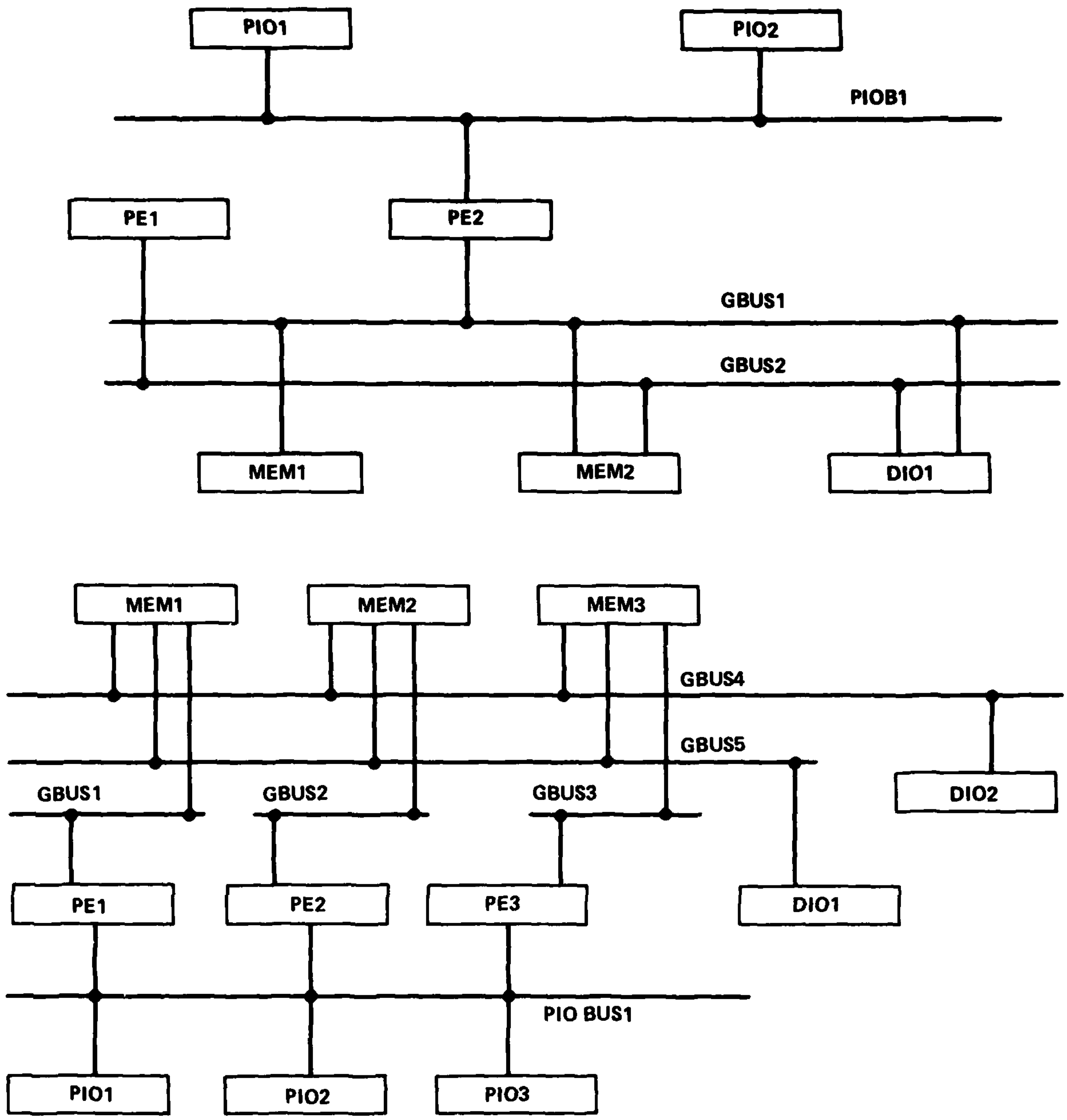


Figure 2. Examples of GCSS Hardware Diagrams

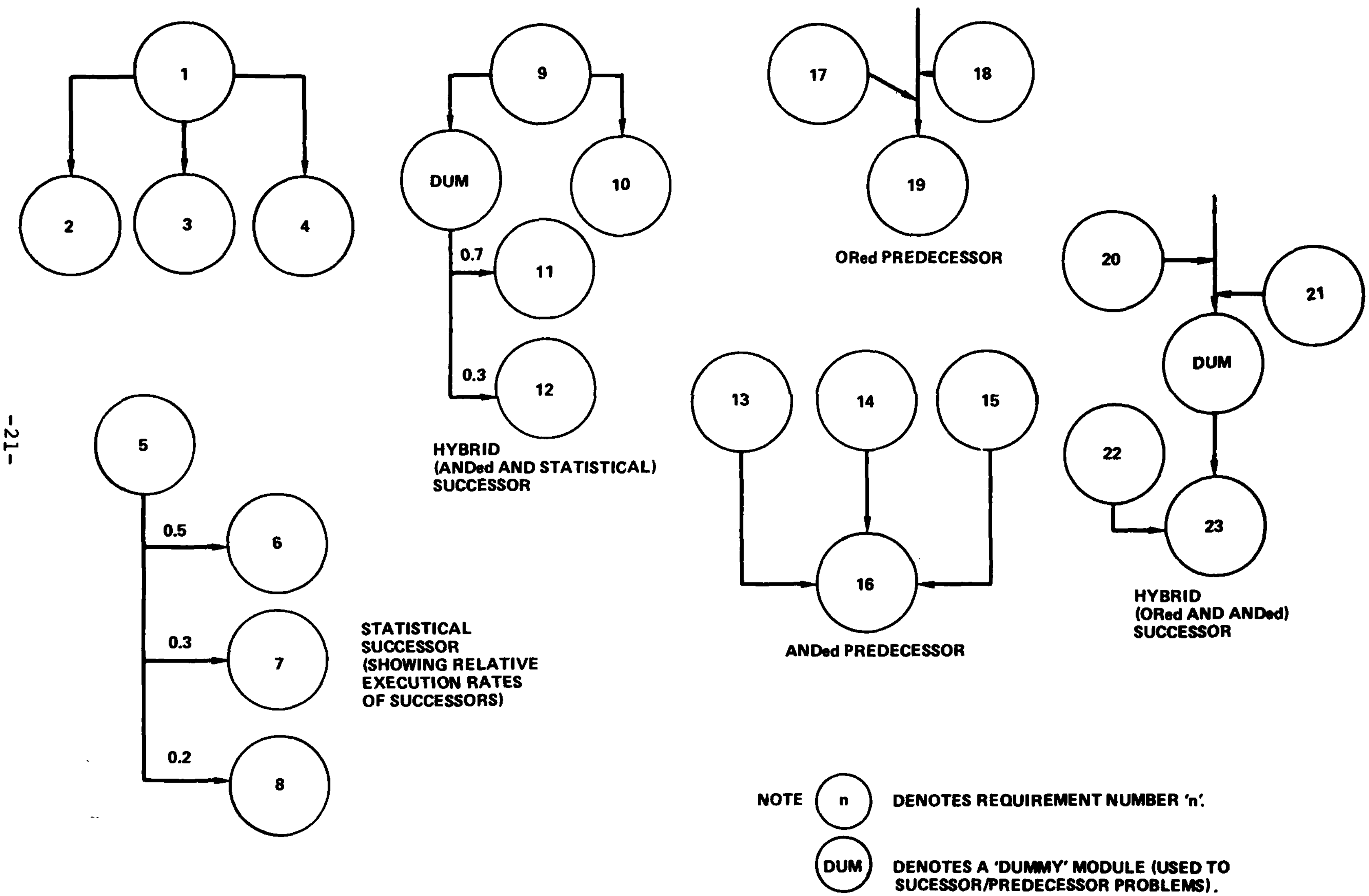
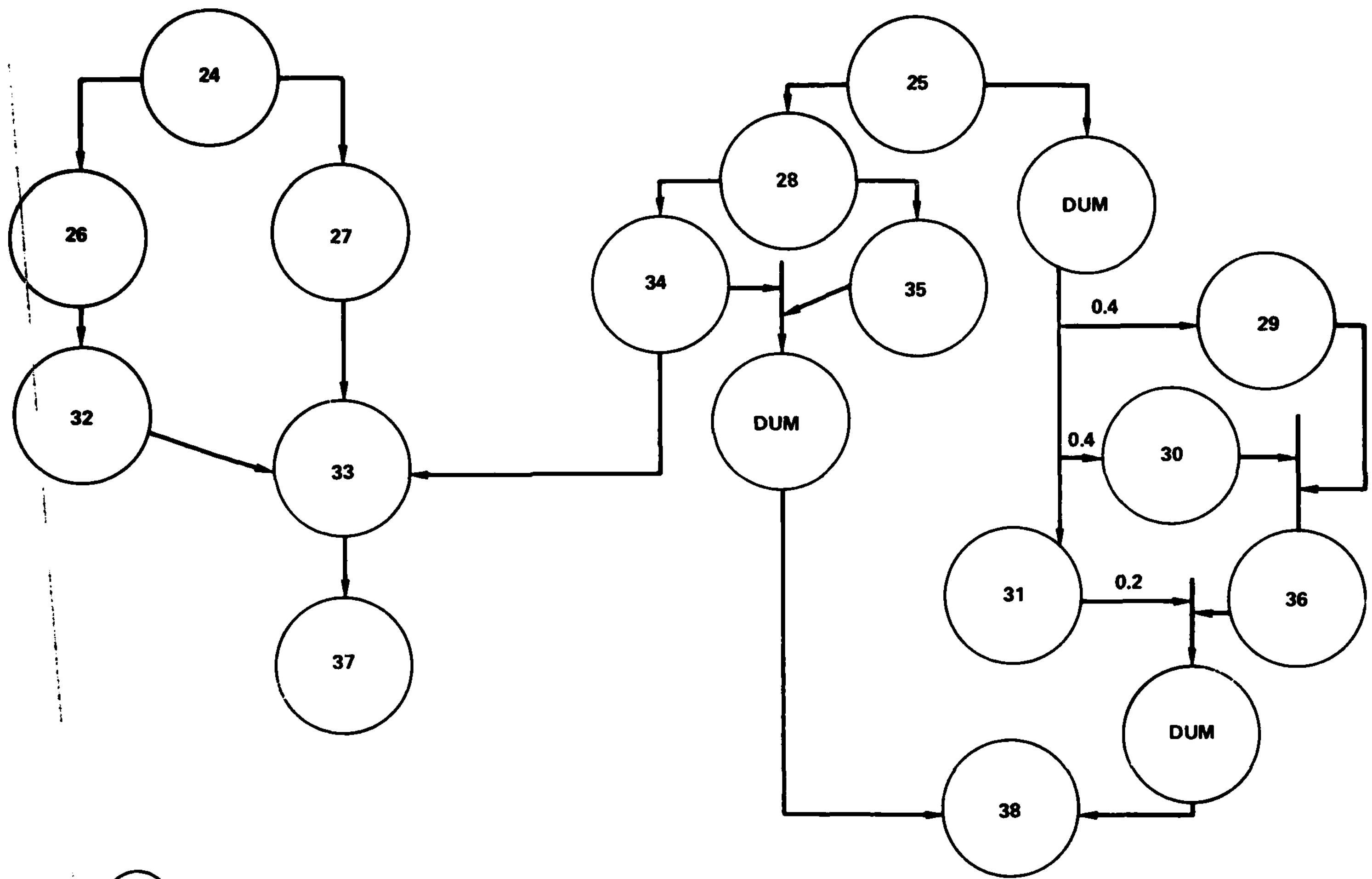


Figure 3. Requirements Chaining



NOTE: (n) DENOTES REQUIREMENT NUMBER 'n'.

(DUM) DENOTES A 'DUMMY' MODEL (USED TO CONTROL LOOPING AND TO RESOLVE SUCCESSOR/PREDECESSOR PROBLEMS).

Figure 4. Example of GCSS Requirements Chaining

ANDed predecessor means all modules chained to a particular module must be completed prior to initiation of that module.

ORed predecessor is the condition where the completion of either module is sufficient to initiate the succeeding module.

While GCSS provides ample diagnostic capability to the user, the interactive system employs a unique capability that allows the user to validate the modeled system configuration prior to execution. Through use of the interactive graphics display terminal, GCSS displays back to the user block diagrams and chaining diagrams of the type that appear in Figure 1 and Figure 3, which characterize the model's physical and functional relationships. This data is derived from the user input data. Thus the user is able to validate the model to be executed against the intended physical and functional relationships prior to actual execution.

System Attributes - GCSS

Performance - Few data points are available that would support a valid assessment regarding performance. An example provided to SASC showed seven seconds of CPU time required to process two-seconds of simulated time. The model being considered consisted of seven PE's, seven MD's, fourteen GBUS's, two PIOB's, two PIOD's, and four DIOD's.

Simulation Capacity - GCSS has the capacity to simulate 1024 each of PE's, MEM's, PIOD's, DIOD's and their bus interconnections.

Ease of Use - GCSS requires no special programming skills to describe or execute a model. Model development times are essentially a function of the complexity of the model being simulated and the desired level of planned analysis.

Input Language - At present GCSS uses no special HOL for the user to interface with the system. The thirty-one UIV's have an established protocol that is relatively cumbersome to handle for large models. However an

effort is currently underway to enhance the user interface by developing a "Concise Architecture Description" language in APL. Upon completion, it is anticipated that model build times can be reduced by approximately fifty percent.

Flexibility - GCSS offers flexibility in terms of the types of data processing architecture that can be modeled and the various levels of analysis that can be conducted. Types of architectures include multiprocessing, distributed processing, and multicomputer. Analysis levels can be conducted at higher functional levels, detailed instruction levels or combinations of both.

Outputs and Reports - A comprehensive set of statistical output reports is available to the user regarding throughput, queue sizes, etc., for each user defined model element. These are provided via formatted data on program listings upon completion of the simulation run. Graphical output is currently limited to the previously mentioned system block and chaining diagrams used to validate user input data and only on the interactive system.

Accessibility - GCSS compatibility with available AFWAL/AAAS resources exists only in terms of the host computer system (CDC 6600). Software resources are not compatible. The CDC system at NADC uses the KRONOS operating system, whereas the local WPAFB CDC system runs under NOS/BE. Also, GCSS is written in SIMSCRIPT I.5 which is not supported locally. These incompatibilities create large portability problems. However, the NADC system is tied into the ARPANET system as is the local WPAFB system. Thus, access to GCSS can be obtained over this network. Accomplishing this interface will require

an access/authorization code to enable access to the NADC system and the hardware elements (i.e., interactive graphics display terminal, hard copy device line printer and modem) necessary to support the transactions.

Documentation - There is no documentation that is representative of GCSS as it is currently configured. This applies to description of the simulator as well as user manuals. A batch system user manual is available; however, if AFWAL chooses to use GCSS, an interactive system user manual will be required.

4.2 DATA SYSTEM DYNAMIC SIMULATOR (DSDS)

4.2.1 INTRODUCTION TO DSDS

The DSDS was developed by General Electric Corporation for the National Aeronautical and Space Administration (NASA), under the direction of the Marshal Space Flight Center (MFSC), Huntsville, Alabama. The system was designed to model and simulate large data processing and communication systems and their respective operational and cost characteristics. The system features a library of preprogrammed models characterizing about 150 elements used in communications and data processing needs in the 1985 to 1990 time frame.

The DSDS is operational in two modes. A batch version of DSDS was developed on an IBM 360/75 computer. An interactive system is being developed on a PRIME 400 system, and is basically IBM 360 compatible. The batch version is available to outside agencies. The system is accessible; in fact, ESD/MITRE is currently installing the batch version of DSDS on their IBM 370 computer to conduct in-house assessments of the system.

The system is programed using GASP IV for system and control functions and FORTRAN IV for lower level functions. Thus rehosting efforts should not pose insurmountable problems. A more detailed discussion of the system is presented in the following section.

4.2.2 DSDS DESCRIPTION

Source

National Aeronautical Space Administration, Marshall Space Flight Center, Huntsville, Alabama

Contact

Mr. John R. Piner - NASA/MSFC

Mr. Norman F. Geer - GEC

Reference Data

NASA "Training Manual For Data Systems Dynamic Simulator" General Electric Company, Space Division, Huntsville, Alabama, September 1, 1978.

NASA "Study to Establish Models and Simulation for Data Systems, Volume 2 Users Manual", General Electric Company, Space Division, Huntsville, Alabama, July 28, 1978.

NASA "Spacelab Simulation, SL-1 Model Implementation Report", General Electric Company, Space Division, Huntsville, Alabama, March 1979.

Application

Communication and Data Processing Systems.

Description

The DSDS is a discrete event driven simulation system designed for performing design analysis of satellite communications and ground data processing systems. The system is programmed in FORTRAN IV and uses a modified version of GASP IV for system level functions and control. The system is operative in a batch mode on an IBM 360/65 and an interactive mode on a dedicated PRIME 400 system. DSDS features modeling capabilities that include hardware, software, human operations, and system characteristics that include timing, control, sizing, external influences, and cost.

The DSDS model development is based upon basic modeling entities called Data System Element Models (DSEMs). These are FORTRAN representations of real systems elements such as sensors, peripheral devices, processors, software tasks, etc. When

developing models, DSEMs are interconnected to form complete system models by directing the output from one DSEM to the input terminal of another. Thus subsets of DSEMs are interconnected to characterize a complete system model which is thus tailored to the user's needs. Figure 5 illustrates the concept of interconnecting DSEMs.

The DSDS maintains a model library of approximately 150 DSEMs. These models are representative of system elements and functions common to the current application of satellite communications and ground data processing systems. Table 3 lists many of the DSEMs currently in the library. In the library, as indicated by the tables, DSEMs are maintained at three levels of detail - Global, Intermediate and Subsystem, where Global level DSEMs are the least detailed and Subsystem level DSEMs are the most detailed. DSDS allows the interconnection of DSEMs of different levels. This feature results in savings of modeling and computer execution times when using Global level DSEMs because they execute their respective elements at the level of detail modeled.

At the system level, DSDS is comprised of six core models that perform the functions of control, event scheduling, statistical data gathering and reporting. Figure 6 shows the core models and their interrelationships. Descriptions of their functions are discussed below.

1) Mission Model (MM) - MM is the DSDS simulation driver. It generates the sequences of events which results in data generation, routing and control based upon the users source input data.
2) Throughput/Resource Utilization Model (TRUM) - This is the simulation's mathematical and logic model. It represents the functional operation of the modeled system and is the mode through which all data flows.
3) Device Utilization Model (DUM) - This is the simulation systems instrumentation element. The DUM provides the reporting functions for the simulation.

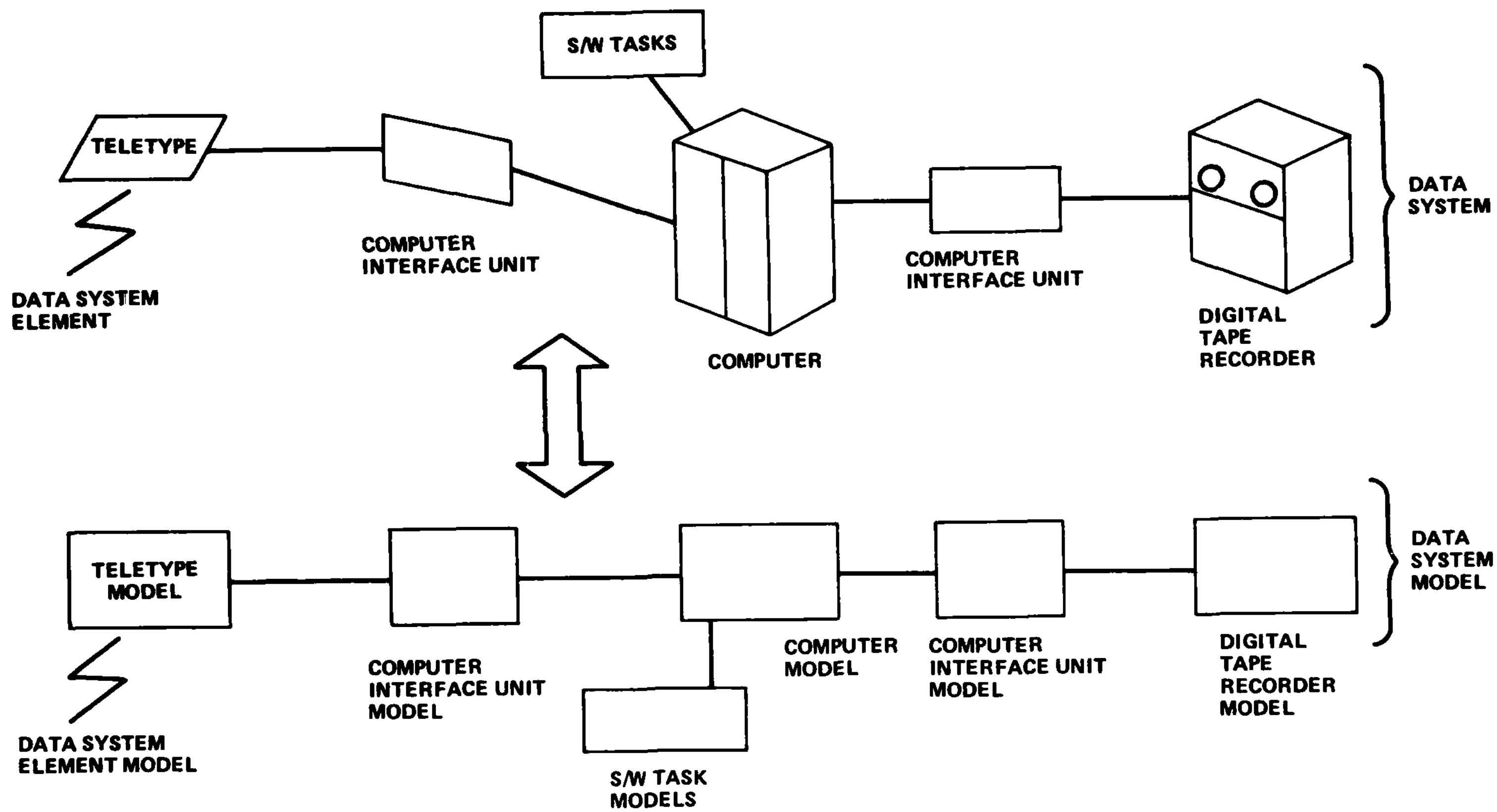


Figure 5. The Concept of Interconnecting Models of Elements Is the Same as That of Connecting Real Elements

TABLE 3. EXAMPLES OF AVAILABLE DSDS DATA SYSTEM ELEMENT MODELS

GLOBAL LEVEL DSEMS

AOIPS - Atmospheric and Oceanographic Information Processing System
Digital Image Processing
*Data Management Element
Image Processing Element
Low Cost Ground Station
NASCOM - NASA Communication Network
Operations Control Center
Prime International Ground Station
Project Office
SEASAT Satellite
Prime Ground Station (STDN)
TDRS Ground Station
TDRS Satellite
Wide Band Data Relay Satellite

INTERMEDIATE LEVEL DSEMS

Data Transfer

*Antenna
Data Coupler
De-Modulator
*Data Bus
*Modulator
*RF Receiving Unit
Transmission Line
RF Transmitting Unit
Transmission Path

Human Processes

Digital Image Assessment
End of Shift Scheduler
Mail Service
Production
Personnel Scheduler
Quality Control
Sorter
User

Special Equipment & Processes

*Experiment
Maintenance (Equipment)
*Sensor
*General Purpose Timing Unit
Tracking Subsystem

Data Storage

Analog Tape Recorder w/Control
Digital Tape Recorder w/Control
Loop Recorder w/Control
Multi-Speed Analog Tape Recorder
Multi-Speed Digital Tape Recorder
Multi-Speed Video Tape Recorder
Strip Chart Recorders
Video Disc with Control
Video Tape Recorder w/Control
- -

Data Switching, Manipulation & I/F

*Analog to Digital Converter
*Digital to Analog Converter
Decoder

Display, Monitor and Input

Display (Monitor)
*General Purpose Data Terminal/ Initiator
Hard Copy
Interactive Console
Line Printer
Teletype and Keyboard
Data Format Conversion Unit (General)
De-Multiplexer
Encoder
*General Purpose Signal Processor
*Low Pass Filters (Video, Analog, Digital)
*Multiplexer
Pre-Modulation Processor
RF Amplifier
Signal Conditioner (General)
Signal Distributor
Signal Switching Unit
Synchronizer (Buffer)
Telementry Format Generator

NOTE: DSEMs marked with an asterisk(*) are those of special interest to AFAL.

TABLE 3. EXAMPLES OF AVAILABLE DSDS DATA SYSTEM ELEMENT MODELS (Continued)

SUBSYSTEM LEVEL DSEMS

Batch applications Program	Mark Sense Reader (Computer Peripheral)
Card Punch (Computer Peripheral)	Magnetic Tape Unit
Card Reader (Computer Peripheral)	Paper Tape Punch (Computer Peripheral)
Computer Interface Unit	Paper Tape Reader (Computer Peripheral)
CPU	Random Access Mass Storage
Disc	*Real Time Program
Drum Unit	Sequential Access Mass Storage
Display (Computer Peripheral)	*Software
Hardcopy Unit (Computer Peripheral)	*Special Purpose Program
Interactive Console (Computer Peripheral)	Transaction Processing Program
Line Printer (Computer Peripheral)	Teletype (Computer Peripheral)
*Memory	

MODEL ORIENTED & CONTROL DSEMS

ANDS ATRIB 5	Rate Switch
MM Interface (Control & Routing Logic)	*RF Constraints
Periodic Data Saver for Restart	Selector Switch
*Communications Link	Set Switch
*General Purpose Data Generator	Single Pole Double Throw (Single Control)
*Data Sensor	Single Pole Double Throw (Dual Control)
*Fixed Time Delay	Stepping Switch
Demand Queue	Throughput Finish Line
Duplicator	Throughput Start Line
INVERTS ATRIB 5	Initialization Control Generator
ORS ATRIB 5	
*Random Routine Switch	

NOTE: DSEMs marked with an asterisk (*) are those of special interest to AFAL.

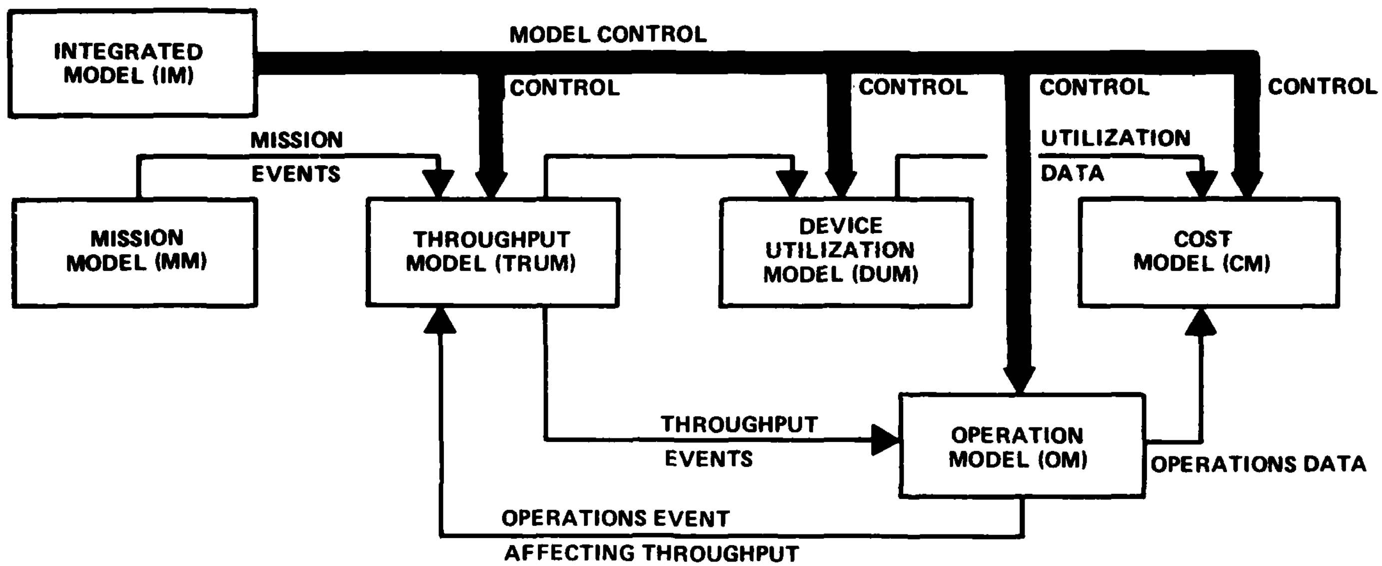


Figure 6. Organization and Interrelationship of Core Models

4) <u>Operations Model (OM)</u> - This model performs the scheduling and allocation of human resources required to support the flow of data in the system.
5) <u>Cost Model (CM)</u> - This is the system's cost estimator. It provides cost estimates of system acquisition and operation.
6) <u>Integrated Model (IM)</u> - This is the simulation executive. IM provides the overall control functions for the other five models.

Performance measurements obtained from DSDS simulations are made possible by logic contained in each DSEM. This logic duplicates the throughput, utilization, operational support requirements, computer resource requirements and cost characteristics of the element represented by the DSEM. The logic is triggered by definitions of the element's operational and interconnection characteristics assigned by the modeler. DSDS requires a set protocol for each DSEM used in a model. Thus, for each DSEM, there is a unique set of parameters that characterize its operation.

The DSDS maintains an up-to-date users' manual which provides the model developer with descriptions of all available DSEMs in the library. Each DSEM has an associated set of parametric codes that enables the user to define the characteristics of the modeled device as it pertains to:

- Throughput - PARTRM
- Device Utilization - PARDUM
- Operations - PAROM
- Cost - PARCST

In addition to the above, the user is prompted about DSEM use with operational notes, special instructions, and block diagrams. Figure 7 contains the block diagram for an Intermediate Level DSEM of a Loop Recorder with Control. This is the basic element from which the other DSEM elements are derived.

The cost data for the DSEM, diagrammed in Figure 7, is as follows:

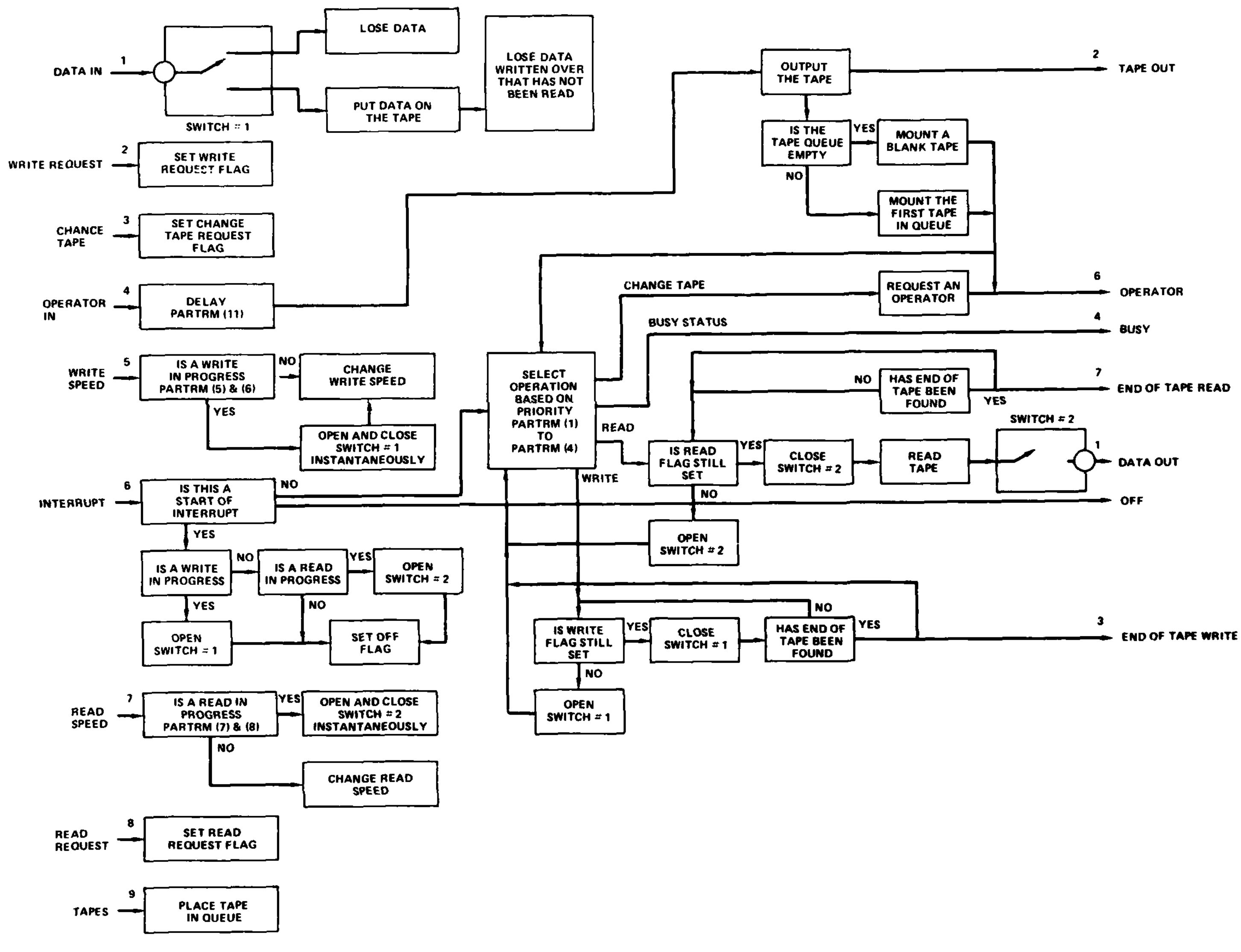


Figure 7. Intermediate Level DSEM of Loop Recorder With Control: Internal Logic Diagram (Concluded)

PARAMETER	DESCRIPTION	UNITS	RANGE	DEFAULT
1	SUBCLASS IDENTIFIER 1-FLIGHT, 2-GROUND	WHOLE NUMBERS	0, 2.0	0.0
2	USER-DESIGNATED SPECIFIC COST	DOLLARS	OPEN	10K
3	FIRST DRIVING COST (EXEC. TIME)	SECONDS	30-30K	5K
4	SECOND DRIVING COST (WEIGHT)	KGM	1-20	3
5	THIRD DRIVING COST (RESERVED FOR FUTURE USE)	N/A	N/A	0.0
6	FOURTH DRIVING COST (RESERVED FOR FUTURE USE)	N/A	N/A	0.0
6A	OPTIONAL-MULTIPLE USE KEY	NEGATIVE WHOLE NUMBERS	-1.0 to -9.0	0.0

Similarly, the parametric data for the DSEM in Figure 7 is displayed below:

PARAMETER NO.	USER INPUT	DEFAULT VALUE	DESCRIPTION
Thruput Model			
PARTRM(1)	YES	1.	PRIORITY FOR CHANGE TAPE
PARTRM(2)	YES	2.	PRIORITY FOR READ
PARTRM(3)	YES	3.	PRIORITY FOR WRITE
PARTRM(4)	YES	7.	WRITE RATE - LOW (IPS)
PARTRM(5)	YES	15.	WRITE RATE - HIGH (IPS)
PARTRM(6)	YES	7.	READ RATE - LOW (IPS)
PARTRM(7)	YES	15.	READ RATE - HIGH (IPS)
PARTRM(8)	YES	15.	LENGTH OF TAPE (FT.)
PARTRM(9)	YES	90.	TIME TO CHANGE TAPE (SEC.)
Device Utilization Model			
PARDUM(1)	YES	0.	CAPACITY
PARDUM(2)	YES	0.	IS TIME POINT DATA DESIRED, 0-No, 1-YES
Operations Model			
PAROM(1)	YES	1.	HOW MANY SUPPORT SKILLS
PAROM(2)	YES	2204.	OPERATING SKILL #1
PAROM(3)	YES	1.	HOW MANY OF OPERATING SKILL #1 ARE REQUIRED
PAROM(4)	YES	8107.	SUPPORT SKILL #1
PAROM(5)	YES	.05	USE RATE SUPPORT SKILL #1 (HRS/HR)
PAROM(6)			SUPPORT SKILL #2
PAROM(7)			USE RATE FOR SUPPORT SKILL #2 (HRS/HR)
PAROM(8)			SUPPORT SKILL #3
PAROM(9)			USE RATE FOR SUPPORT SKILL #3 (HRS/HR)

The textual material, which completes the data needed for the DSEM in Figure 7, consists of Operational Notes and Special Instructions.

<u>Operational Notes</u> - This tape recorder model is capable of writing data (records), reading data (play back), changing tape, and stopping for maintenance. These operations are given priorities by the user except for maintenance which always has top priority. If the recorder is in the process of reading, and the recorder is shut down for maintenance, the read operation is immediately halted. The same is true for the write operation. The change tape operation will ignore maintenance and will finish before the tape recorder is shut down. If the recorder is busy with one of these operation other than maintenance and receives control for other operations of a higher priority then the controls are queued by priority and must wait for the present operation to finish. Data is lost whenever data tries to enter the recorder and no write command has been given or the recorder is busy doing something else. Data is also lost if data on the tape that has not been read is written over.

<u>Special User Instructions</u>

1. Status Outputs
   A. End-of-Tape Write - If the recorder is writing and finds the end of the tape then an "ON" control is output followed by an "OFF" control one microsecond later.
   B. Busy - When the recorder begins reading or writing an "ON" control is sent out. If at the end of the operation the next request in the queue is the other operation then no control signal is sent out. When all requests for reading and writing have been satisfied or one of these operations is interrupted, then an "OFF" control is sent out.

C. OFF - An "ON" control is sent out when maintenance starts and an "OFF" control when it is done.

D. End-of-Tape Read - If the recorder is reading and finds the end of the tape then an "ON" control is output followed by an "OFF" control one microsecond later.

2. The smaller the number entered for priority, the higher the priority. The highest priority that can be entered is 1.

3. When an "ON" control is received in either speed select, the recorder switches to the higher speed. An "OFF" control returns the recorder to low speed.

4. A change tape request will cause the recorder to request personnel. If the oeprator request line has been given a zero for its connection then the recorder will assume it has a person dedicated to it and go ahead with changing the tape. If no tapes have been received then a blank tape is mounted.

Utilization Definition - The utilization equals 100% when the recorder is showing busy on its status line output and 0% the rest of the time.

Figure 8 is the block diagram for a Subsystem Level DSEM of a Computer Interface Unit. This DSEM has no associated cost data, and its parametric data is as follows:

PARAMETER NO.	USER INPUT	DEFAULT VALUE	DESCRIPTION
Thruput Model			
PARTRM(1)	YES	4.	RANDOM NUMBER CODE
PARTRM(2)	YES	1.	RANDOM NUMBER FIRST PARAMETER
PARTRM(3)	YES	.5	RANDOM NUMBER SECOND PARAMETER
PARTRM(4)	YES	1.5	RANDOM NUMBER THRID PARAMETER
PARTRM(5)	YES	.0833	RANDOM NUMBER FOURTH PARAMETER
PARTRM(6)	YES	0.	NOT USED
PARTRM(7)	YES	526	MAXIMUM BITS IN BUFFER
PARTRM(8)	YES	1000.	OUTPUT DATA RATE
PARTRM(9)	YES	1000.	MAXIMUM RATE FOR INPUT (BPS), ALSO THE NUMBER OF BITS PROCESSED IN THE RESOURCE TIME SPECIFIED IN PARTRM(13), (16), ETC.

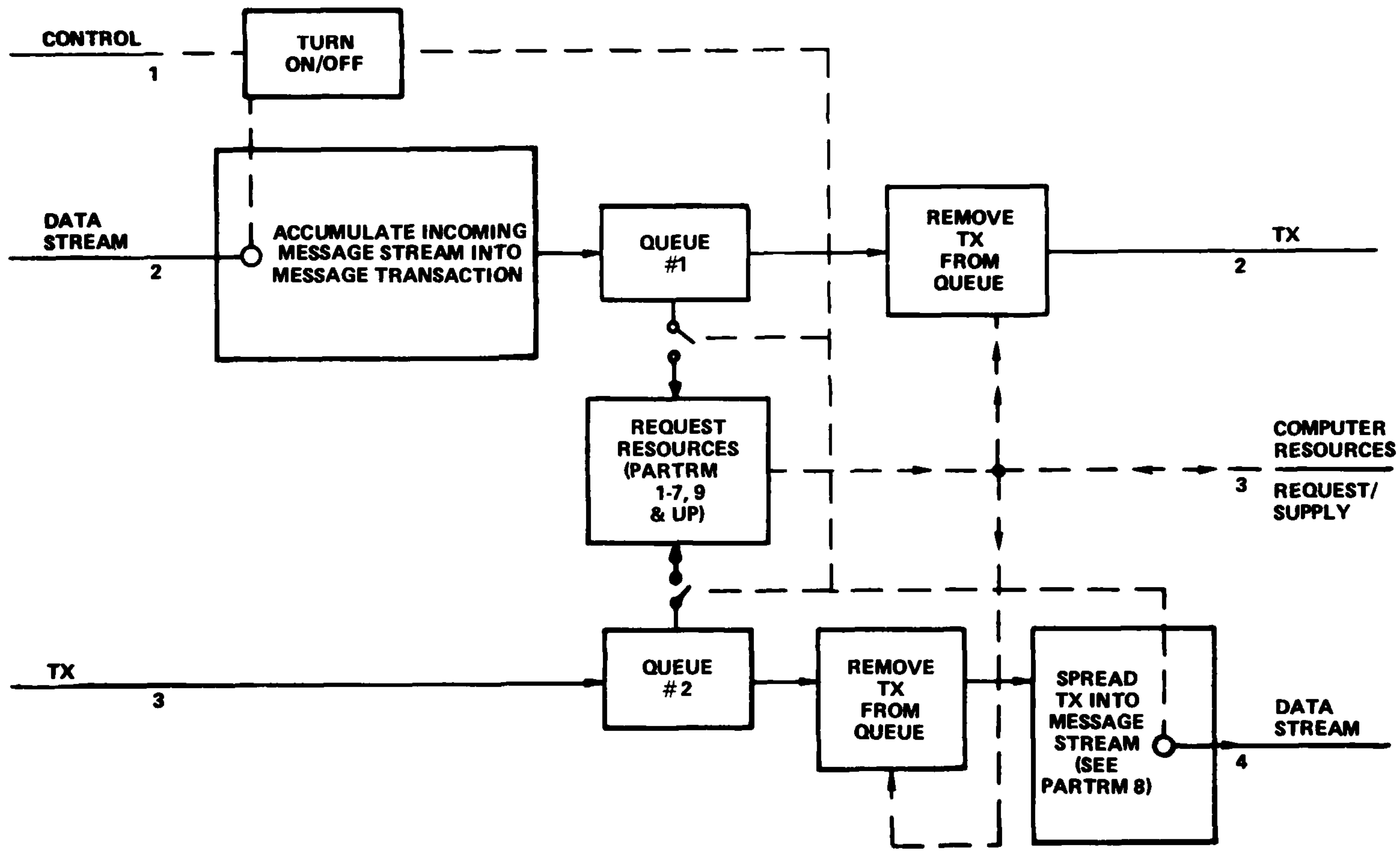


Figure 8. Subsystem Level DSEM for a Computer Interface Unit

PARAMETER NO.	USER INPUT	DEFAULT VALUE	DESCRIPTION
Thruput Model			
PARTRM(10)	YES	4.	NUMBER OF RESOURCES
PARTRM(11)	YES	100.	FIRST RESOURCE REQUIREMENT CODE
PARTRM(12)	YES	1.	FIRST RESOURCE REQUIREMENT QUANTITY
PARTRM(13)	YES	1.	FIRST RESOURCE REQUIREMENT TIME (SECS.)
PARTRM(14)	YES	5.	SECOND RESOURCE REQUIREMENT CODE
PARTRM(15)	YES	1000.	SECOND RESOURCE REQUIREMENT QUANTITY
PARTRM(16)	YES	1.	SECOND RESOURCE REQUIREMENT TIME (SECS.)
PARTRM(17)	YES	3.	THIRD RESOURCE REQUIREMENT CODE
PARTRM(18)	YES	1000.	THIRD RESOURCE REQUIREMENT QUANTITY
PARTRM(19)	YES	1.	THIRD RESOURCE REQUIREMENT TIME (SECS.)
PARTRM(20)	YES	106.	FOURTH RESOURCE REQUIREMENT CODE
PARTRM(21)	YES	1.	FOURTH RESOURCE REQUIREMENT QUANTITY
PARTRM(22)	YES	1.	FOURTH RESOURCE REQUIREMENT TIME (SECS.)
Device Utilization Model			
PARDUM			NONE
Operations Model			
PAROM			NONE

The textual and final portion of the DSEM in Figure 8 contains instructional material.

Operational Notes

1. The CIU will shutdown for maintenance when an off control Event Type 2 arrives at Pin 1. It will start up when an on control Event Type 1 arrives at Pin 1.
2. When a Message Stream arrives at Pin 2, it will be accumulated into a message transaction (TX). The TX will be placed in Queue #1 in order of arrival. If a complete message has entered the CIU003, resources are requested.
3. When the resources are supplied, the TX is removed from Queue #1 and output on Pin 2. After a delay equal to the maximum resource 'use time' as defined under Special User Instructions, the resources are returned to the CPU003. If Queue #1 is not empty, the resources are requested for the event in the top of Queue.

4. When a TX arrives at Pin 3, it will be placed in Queue #2. If Output #1 is not busy, resources are requested for processing.
5. When the resources are supplied, the TX is spread into a Message Stream.

Special User Instructions

1. The resource Requirements Table (RRT) is entered in PARTRM(10). Each Resource definition requires 3 parameters: (1) Resource Code, (2) Resource Requirement Quantity, and (3) Resource Requirement Time. Refer to "Resource Code Table" Appendix #3 for the code and quantity definitions of each resource.
2. The maximum number of bits to be buffered (e.g., on the Disc) limits the quantity of data that may be stored in the CIU003. It is called Maximum Bits in Buffer and entered in PARTRM(7).
3. The Bit Rate for the output message stream is entered in PARTRM(8).
4. The Random Number distribution code and parameters are entered in PARTRM (1 thru 5). The 'use time' of each resource is determined by (Random Number) * (Number of bits in incoming message transaction) * (resource time in PARTRM(13), (16), etc.) † PARTRM(9).

Utilization Definition - The utilization is determined by the CPU003.

Since all pertinent performance information is programmed into each DSEM, DSDS is able to provide performance measurements from simulations at system and element levels using any set or combinations of DSEMs. Performance measurements that are made available are:

- Message transit time
- Average transit time
- Transit time between any two points
- Mean throughput, utilization and transit time per element

- Storage statistics and percent utilization
- Tabular throughput timeline
- Computer resource utilization summary
- Plot of storage device full

Other reports include personnel skills and allocation requirements from the operations model and system and element cost estimates from the cost model.

It is significant to note that DSDS has no operational constraints or limits with regard to the number of DSEMs that can be assigned to the library aside from the host system physical storage limits. New DSEMs are still being defined and added to the system. This capability makes DSDS a highly flexible tool with the potential to support a variety of data processing and information processing analysis efforts.

System Attributes

Performance - Typical DSDS performance times when executing to obtain only throughput and device utilization to the subsystem level is approximately 1:4. That is, it takes one second of CPU time to execute four seconds of simulated time. Since DSDS is multi-functional and executes in multiple levels, it is not possible to quantify its performance as a constant.

Simulation Capacity - There are no limits placed on either model size (number of DSEMs) or the number of events that can be simulated assuming that adequate storage is available. Currently a full program load is 1460K bytes. In its present configuration disk storage is used to reduce core requirements during execution. Under control of the executive, core requirements are approximately 350K bytes.

Ease of Use - DSDS is user oriented. The concept of interconnecting DSEMs end-to-end simplifies the model development process. User aids consisting of block diagrams

of DSEMs, labeled inputs, outputs, and identification of element characteristics enhances overall model development and learn times.

Input Language - The DSDS input language form is straightforward and requires no special skills. User inputs are prepared in standard NAMELIST input common to FORTRAN IV. Each DSEM is individually specified within a NAMELIST block in terms of throughput, device utilization and interblock connections. An example is given in Figure 9.

Flexibility - The DSDS exhibits flexibility in areas of:

- Design - The system can simulate almost any data processing configuration that can be conceived provided DSEMs are available.
- Analysis - Models can be characterized and performance measured at three functional levels using any combination of DSEMs. Types of analysis are user options, i.e., cost, throughput, device utilization, or operations.
- Growth - System growth is limited only by physical host system constraints and the ability of personnel resources to develop new DSEMs.
- Rehostability - Since DSDS uses GASP IV or control and FORTRAN IV to implement DSEMs, rehosting should be a relatively straightforward task.

Outputs and Reports - Statistical output reports are available at system level and on the performance of each DSEM in a given model independent of functional level. The reports include queue statistics, throughput, device utilization, operational resource requirements and cost. Graphical outputs on all of the above is available with the

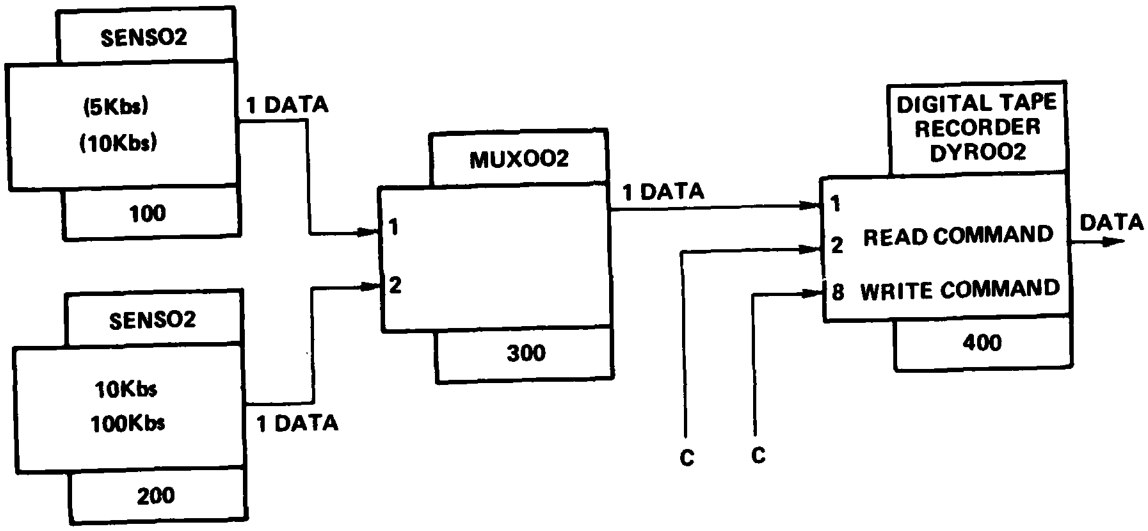


Figure 9. DSEM Describing a Multiplexer and Its Interconnections

interactive system. A sample of DSDS output is shown in Table 4.

Accessibility - Since DSDS uses GASP IV or FORTRAN IV, it can probably be hosted on either the AFWAL DEC 10, or the ASD CDC 6600 computer system. The batch version is available through the University of Georgia's COSMIC library services.

The interactive system is in the final stages of development and is not planned for release to outside agencies until late 1980. In either case, it would be advisable to solicit specialized support from General Electric Corporation to effect the rehosting effort. This support, along with specialized training of prospective users of DSDS, can be obtained from General Electric Corporation.

Documentation - The DSDS is accompanied by a comprehensive set of user oriented documentation. There is a training manual which describes the system's operation and presents sample problems illustrating user input formats and requirements and simulation outputs. Additionally, there is the user's manual which covers the basic design and use of all system functions and includes a complete and current dictionary of all DSEMs.

4.3 DESIGN ANALYSIS SYSTEM/DISTRIBUTED DATA PROCESSING MODEL (DAS/DDPM)

4.3.1 INTRODUCTION TO DAS/DDPM

The DAS/DDPM is a proprietary simulation and modeling tool developed by Hughes Aircraft Company (HAC), Fullerton, California. The system is a product of evolving simulation and modeling technology development fostered by in-house research and development needs.

TABLE 4. DSDS OUTPUT SAMPLE

REPORT DUM-1 TRANSIT TIME FOR STARTS AND ENDS OF MESSAGES

MESSAGE		ORIGINATION	TIME		TRANSIT TIME			TRANSIT TIME TO USER OR WHERE LOST			ORIGIN		GROUND		USER OR WHERE LOST		LONGEST TIME IN QUEUE			
NO.	SUB	DATE	HRS/		TO GROUND						FACILITY		FACILITY		FACILITY		FACILITY		DURATION	
S/E	NO.	MO/DY/YR	MIN	SEC	HRS	MIN	SEC	HRS	MIN	SEC	SET	BLOCK	SET	BLOCK	SET	BLOCK	SET	BLOCK	HRS	MIN
73E	1	6/ 8/78	1111	11.94	0	0	0.0	0	0	1.000	0	201.00	0	0	0	290	.0	0	0	0.0
74S	1	6/ 8/78	1159	59.94	0	0	0.0	68	0	0.063	0	200.00	0	0	0	590	0	0	0	0.0
74E	1	6/ 8/78	1214	24.94	0	0	0.0	67	45	35.000	0	520.00	0	0	0	590	0	0	0	0.0
74S	2L	6/ 8/78	1214	24.9[illegible]	[illegible]	0	0.0	0	0	0.0	0	520.00	0	0	0	520	0	0	0	0.0
74E	2L	6/ 8/78	1215	11.[illegible]	[illegible]	[illegible]	0.0	0	0	1.001	0	200.00	0	0	0	520	0	0	0	0.0
75S	1	6/ 8/78	1159	59.94	0	0	0.0	0	0	1.000	0	201.00	0	0	0	290	0	0	0	0.0
75E	1	6/ 8/78	1215	11.94	0	0	0.0	0	0	1.000	0	201.00	0	0	0	290	0	0	0	0.0

The Number of Lost Message Parts Which Had Greater Than 0 Data Bits = $\boxed{9}$

The Number of Lost Message Parts Which Had 0 Data Bits =

Since Every 4 Messages Not Fill Tape, There Are 9 Tapes Generated.

The DAS is promoted as a concept and a simulation and modeling development support tool. As a concept, it supports operation analysis, data processing design analysis and software design analysis. As a tool it provides a sophisticated interactive graphics-oriented interface design to enhance the users' interface. The DDPM is a distributed data processing model which contains a library of models representing the more common components found in distributed data processing systems.

The DAS/DDPM is programmed in Product Line Simulation (PLS) 470/v6 computer and is used interactively through Time Sharing Options (TSO) via a HP2648A intelligent graphics terminal. A version of the system is being developed by Hughes for ESD/MITRE under the name of AISIM (Automated Interactive Simulation Model). Additionally, another version of the system is being proposed to RADC to support their C^3 analysis needs. Both activities involve translating the software from PLS to SIMSCRIPT II.5. Thus the system is accessible for those parties wishing to purchase the capability. A detailed discussion of the system is presented in the following section.

4.3.2 DESIGN ANALYSIS SYSTEM/DISTRIBUTED DATA PROCESSING MODEL DESCRIPTION

Source

Hughes Aircraft Company, Ground Systems Group, Fullerton, California. DAS/DDPM is a proprietary product of Hughes Aircraft Company.

Contacts

Mr. John Camp

Mr. Robert Jones

Reference Data

Mr. R. Willis "Overview of Hughes' Design Analysis System (DAS)"

Mr. W. Austell "Overview of Hughes' Distributed Data Processing Model (DDPM)"

Note: This data was received in briefing form in a technical interchange meeting held at Hughes in January 1980.

Application

Operations analysis, distributed data processing analysis, software design analysis.

Description

The DAS is a simulation and modeling development support tool. The system is hosted on an AMDHAL 470/v6 and is used interactively through TSO via an HP2648 intelligent terminal. The system is comprised of interactive editors for designer and analyst interfaces, a generalized simulator program and a generalized data base management system.

The DAS architecture is illustrated in Figure 10. The system features diagramatic development of models at the terminal, automatic translation of diagrams to simulation (obviating the need for programming and debugging) interactive variable editing and on-line plotting and documentation of results.

A Design User Interface (DUI) allows model designs to be created graphically at the terminal from a data base of symbols displayed on the terminal. Interactively these symbols are interconnected in a flowchart manner. (Depending upon the data base, the user defines architectures and/or processors). Figure 11, shows an example of an operational process. The symbols on the left are a menu of symbols used to prompt the user while building a process. The flowchart on the right represents the as-built process. Inverse video forms are used to prompt the user in assigning the required parameter data characterizing the operational characteristics for each block. See Figure 12.

During model development, the design information is maintained in a data base for future access by the Analysis User

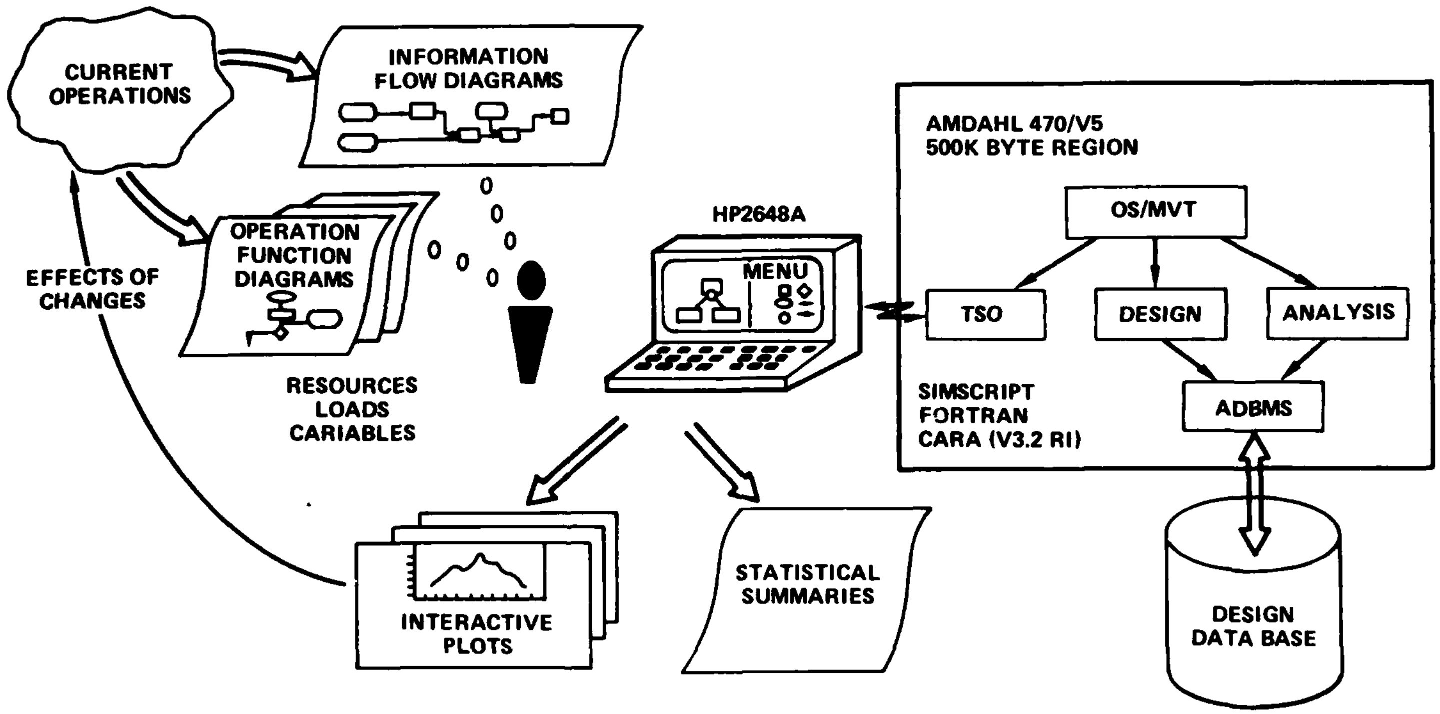


Figure 10. The Design Analysis System Architecture

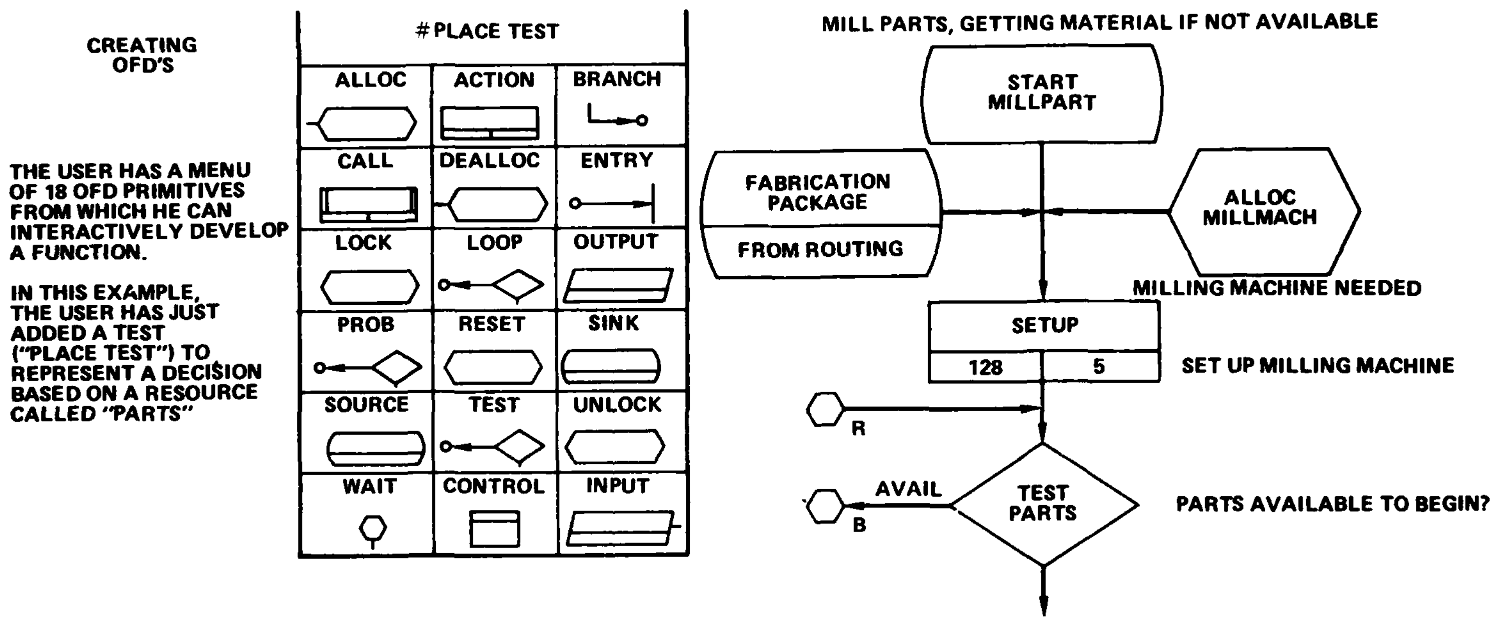


Figure 11. The DAS DUI Supports Interactive Development of Processes and Architectures

CREATING OTHER ENTITIES

BESIDES OFD'S, RESOURCE, VARIABLES, AND SCENARIOS MUST BE GIVEN. THE EXAMPLES SHOWS HOW SCENARIOS ARE DEFINED WITH $\boxed{}$ MEANING "FORM" INPUT

SCENARIO: $\boxed{\text{CONTRACT - 1789}}$ PERIOD LENGTH: $\boxed{\text{4 DAYS}}$

DESCRIPTION: $\boxed{\text{MANUFACTURING LOAD TO DO CONTRACT - 1789 OVER 4 DAYS}}$

LOADS: $\boxed{\text{SHIFT 1, SHIFT 2}}$

SHIFT 1: OFD = $\boxed{\text{ROUTING}}$, TIME = $\boxed{\text{0800}}$, RATE = $\boxed{\text{1/HR/UNIFORM}}$, PRIORITY = $\boxed{\text{1}}$
OFD = $\boxed{\text{MILLPART}}$, TIME = $\boxed{\text{0630}}$, RATE = $\boxed{\text{3/HR/RANDOM}}$, PRIORITY = $\boxed{\text{3}}$
OFD = $\boxed{\text{ASSEMBLY}}$, TIME = $\boxed{\text{0700}}$, RATE = $\boxed{\text{2/HR/CALLED}}$, PRIORITY = $\boxed{\text{5}}$

Figure 12. Inverse Video Forms Prompt the User For Variable Input Data

Interface (AUI) for future updates. Prior to simulation, the data in the data base is automatically translated into an executable form negating the need for programming or debugging.

The AUI is used to evaluate the dynamic performance of the model during and after simulation execution. The General Function Model, a discrete event table driven model performs the simulation. During execution via the AUI, the user has the option to interrogate the system for specific statistical data and to change selected parameters. At the end of the run and at breakpoints during the run, plots of statistical data gathered during execution may be obtained. At the end of the run, plots and tabular data are available. Up to ten plots can be displayed at once. Figure 13 shows types of outputs available via the AUI.

The DDPM is a discrete event simulation mode that supports the investigation of distributed computer system design alternatives. The system features an architecture submodel for defining model interconnection and message routing and a library of preprogrammed submodels of functions of distributed data processing systems. The model is programmed in PLS and is hosted on an Amdhal 470/v6 and is used interactively by timesharing via an HP2648A graphics terminal.

The DDPM consists of an executive driver, support software consisting of a simulator report generator, analysis support software, and the library of submodels. Together these components enable interactive model design, interactive simulation and interactive report generation. The DDPM architecture is illustrated in Figure 14.

Model designs are developed from the library of submodels. Each DDPM submodel is a logical and functional representation of the data processing component it is modeled to represent. Operational characteristics are provided by default or uses specified. Currently existing submodels include:

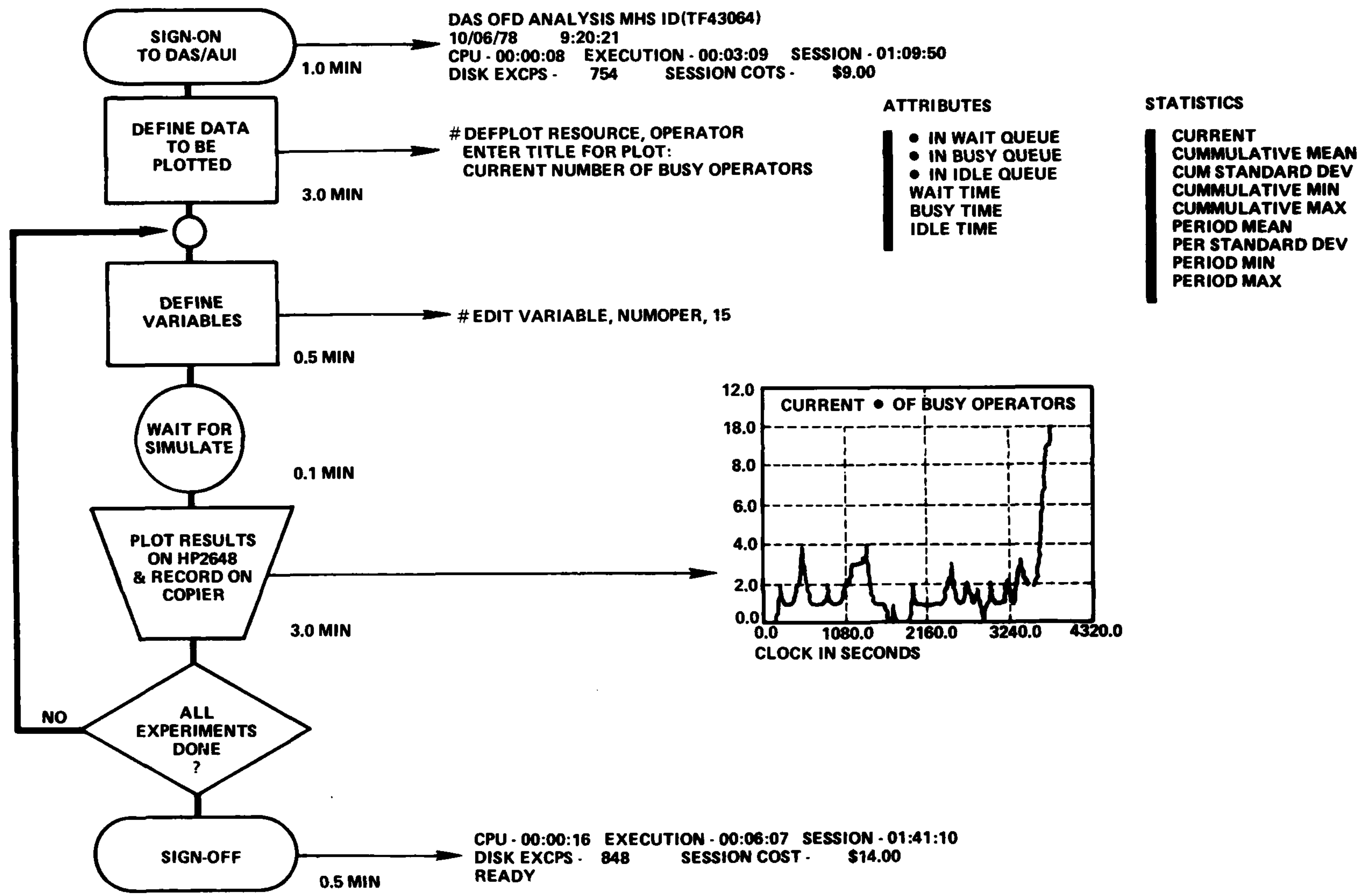


Figure 13. Design User Interface Enables Plots and Tabular Outputs of Model Statistics

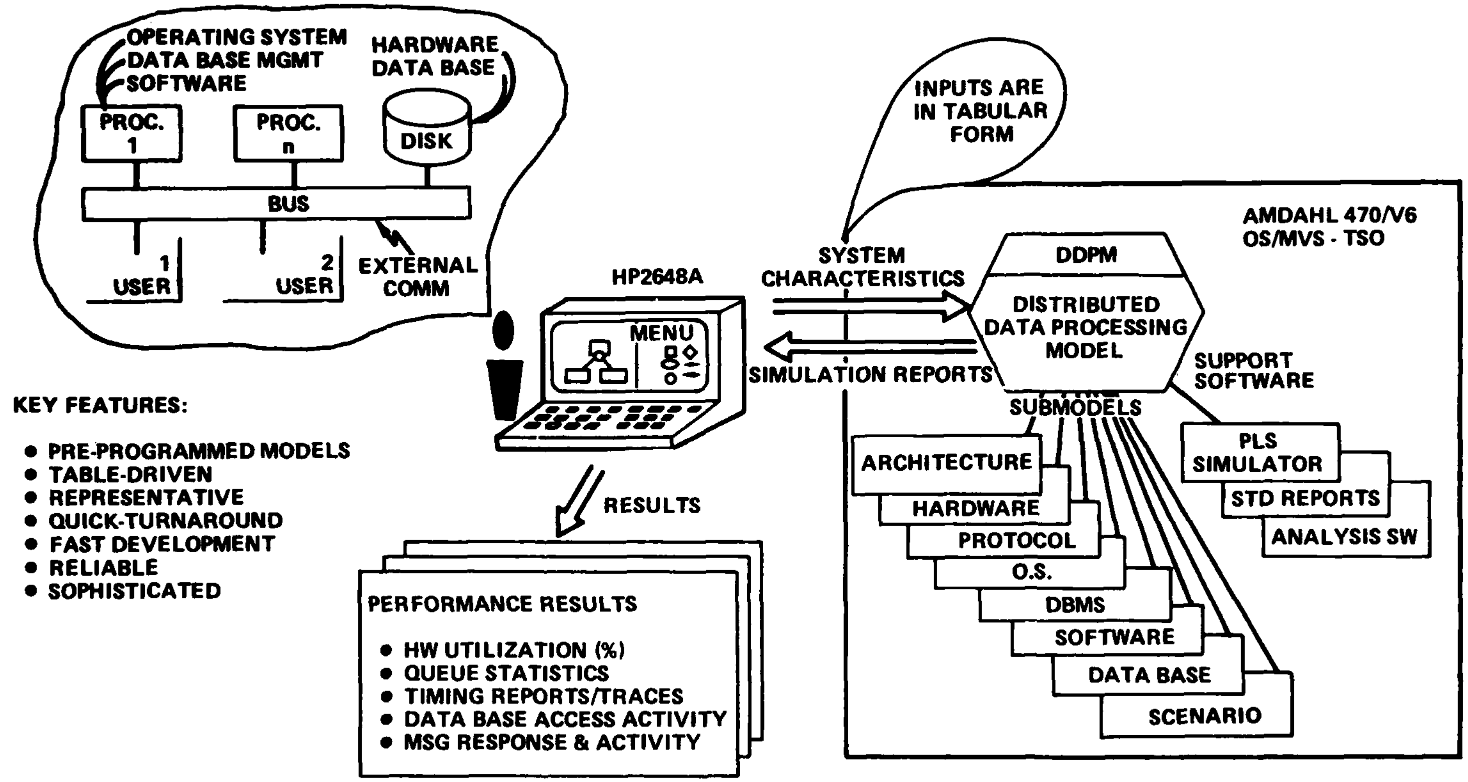


Figure 14. The Distributed Data Processing Model Architecture

- CPU Operating System
- Disk
- Network Control Unit
- Time Rev..ion Multiple Access Bus
- Unibus
- Unibus Protocol Definition
- Architecture and Protocol
- Scenario
- Software
- Data Base Management
- File Management/Data Base

Examples of the types of parameter inputs that characterize some of the submodels' performances are shown in Table 5.

The DDPM provides a variety of statistical reports and summary data. They pertain to:

- Processor Utilization
- Transaction Statistics
- Memory Utilization
- Device Utilization
- Queueing Statistics

These outputs are available in a tabular/listing format.

Another aspect to the DAS/DDPM capability is its ability to interface with a requirements data base. The system has the ability to generate a requirements data base directly readable by the CADSAT (Computer Aided Design and Specification Tool) system. CADSAT can product user specified reports from this output. Further, a version of the DAS/DDPM is able to read a CADSAT generated set of requirements and build a model from that set of requirements.

Finally, the capabilities of DAS and DDPM can be integrated combining the best features of each to provide potential users a fully interactive graphics oriented distributed data processing capability.

TABLE 5. EXAMPLE OF PARAMETER INPUTS TO THE DDPM MODELS

OPERATING SYSTEM	HARDWARE (DISK EXAMPLE)	SOFTWARE DEFINITION
Device Assignment (Node #)	Transfer Rate	Name
Time-Sharing Options:	Interference Rate	Type:
Multi-processing coefficient	Seek Time:	Overlay
Time slice	Overhead	Reentrant
Quantums	Cylinder Head Movement	Reusable
Instruction Execution For:	Rotation Speed	Core Size
Dispatch	Storage Capacity	Buffer Size
Scheduling		Processor Assignment
Time slice monitor		Cyclic Interval
Interrupt processing		Physical Location
Security monitor		Device Number
File manager		Cylinder Number
Core overlay manager		Preemptive or Not
Inter-node communication		Priority
		Logic of Execution:
		Instructions executed
		Decisions
		Calls to other programs
		IO requests

System Attributes

Performance - DAS/DDPM performance times quoted were in the area of 1:15. That is, it takes one second of CPU time to execute 15 seconds of simulated time. This is possible principally because of the speed of the host machine AMDAHL 470/v6 (the AMDAHL is approximately four times as fast as its IBM look-alike).

Simulation Capacity - Simulation capacity was cited in terms of model size of node. The system has the capacity to accommodate models of one-hundred nodes without overflowing data structures or requeueing auxillary storage.

Ease of Use - A judgement regarding ease of use would be speculative since neither DAS or DDPM, are adequately documented. However, the use of interactive graphics and other methods used to aid and prompt the user in the use of these models implies that they are user oriented and easy to use.

Input Language - DDPM inputs are tabular, whereas the DAS inputs are more sophisticated, using Operational Functional Diagrams (OFD) and inverse video forms to prompt the user for parameter values. Both methods are cumbersome, however, and the user will require familiarity with the input formats in order to use the system.

Flexibility - At the system level the DAS has the flexibility to support three types of analysis: Operational design analysis, data processing design analysis, and software design analysis. There are no provisions for selecting various levels of simulations.

Diagnostic - System diagnostics are poor. User must be well versed in all aspects of running either DAS or DDPM.

Output and Reports - Statistical output and reports are available in tabular form or in plots at the terminal. The user has the option to establish breakpoints in a simulation to interrogate the system for tabular or plots of selected data. Up to ten plots may be displayed on the terminal at once. A DAS/DDPM statistical output is shown in Table 6.

Accessibility - Neither system (DAS or DDPM) is compatible with local AFWAL computer services. HAC has hosted the systems on AMDHAL 470's and in particular programmed the DDPM in a proprietary language. On the other hand, HAC is involved in contracted efforts that require translation to rehost the system on an HIS6820 for RADC (proposed) and an IBM 370 for ESD/MITRE. The programs are being translated to SIMSCRIPT II.5. Thus the capability to adapt the capability to other computing environments is available.

Documentation - There is no documentation that adequately describes or guides one on the use of DAS or DDPM. Thus the system is currently run in-house by experts. The MITRE and RADC efforts contracted for documentation.

TABLE 6. PORTION OF SAMPLE OF DAS/DDPM OUTPUT

SOFTWARE EXECUTION SUMMARY

TASK NMBR.	TASK NAME	PRIO RITY	TASK EXCS	% CPU UTIL	ELAPSED TOTAL	EXECUTION TIME (MS) AVG.	MIN.	MAX.	AVG. TIME IN QUEUES (MS) DISPATCH	OVERLAY	SUSPEND	SUSPND BY INT
1	Executive Stub	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
2	Overlay Program To Manager	31	88	1.0	6566.8	74.6	6.7	260.6	6.8	0.0	43.9	0
3	TimeOut Monitor (Time.Critical)	1	396	0.7	232.1	0.5	0.3	3.1	3.5	0.0	0.0	0
4	ADP JCP Fine Hearing Task	8	7	0.0	8.8	1.2	1.1	1.8	31.0	0.0	0.0	0
5	ADP JCP Frequency Task	6	63	2.5	680.5	10.8	7.4	37.2	6.2	0.0	0.0	17
6	ADP PBB Task	7	19	0.0	21.6	1.1	0.0	4.5	4.4	0.0	0.0	12
7	ADP PPI Task	2	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
8	ADP SDR Task	1	104	9.6	2559.6	24.6	18.3	29.7	2.2	0.0	0.0	0
9	ASP TDI/SSI (CP) Task	3	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
10	ADP RDI/SSI (CW) Task	5	117	0.9	301.9	2.5	2.1	5.3	5.6	0.0	0.0	0
11	ADP VDR Search Task	4	102	8.7	2102.2	20.6	17.5	33.4	4.2	0.0	0.0	0
12	ADP VDR Track Task	4	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
13	FOR Ownships Bearing Compensator	10	1	0.0	1.9	1.9	1.9	1.9	50.7	17.9	0.0	0
14	FOR SDR Search Display Formatter	11	1	4.4	12234.4	***.4	***.9	***.4	7.9	50.0	33.7	51
15	IUO Audio Bean Processor	10	51	0.0	21.3	0.4	0.1	1.9	6.7	16.5	0.0	1
16	IUO Compute ADP Parameters	9	39	0.1	27.2	0.6	0.5	1.9	79.9	0.7	0.0	2
17	IUO Enable TOI/SSI Data Input	10	8	0.0	1.7	0.2	0.1	0.4	2.7	0.0	0.0	0
18	IUO Immediate Sonar Request Processor	9	52	0.0	22.3	0.4	0.3	1.6	0.5	0.0	0.0	1
19	IUO SOR/PBB Parameters Switch Process	10	1	0.0	1.3	1.3	1.3	1.3	75.5	74.5	0.0	0
0	IUO Sonar Status Table Processing	9	39	0.7	250.4	6.4	3.9	39.5	6.4	0.0	0.0	19
1	IUO VDR Parameters Switch Processing	10	1	0.0	1.7	1.7	1.7	1.7	132.7	119.3	0.0	0
22	JCP Calculate Harmonic	10	1	0.1	36.5	36.5	36.5	36.5	87.9	151.1	0.0	1

SECTION V

SUMMARY AND RECOMMENDATIONS

5.0 SUMMARY

SASC has identified and described in a preliminary manner three (3) simulation and modeling systems with potential AFWAL/AAAS applicability: GCSS, DSDS, and DAS/DDPM. As a minimum, each system is all-software, operational and models distributed systems. It was also found that AFWAL's interest in fostering its in-house modeling capability is shared by other Air Force agencies. These agencies are currently performing what appears to be the next logical step for AFWAL in its pursuit of a simulation and modeling capability (i.e., detailed evaluation and acquisition).

5.1 RATING FACTORS

GCSS, DSDS, and DAS/DDPM each present different strengths and weaknesses when evaluated by the study criteria. This is shown in Table 7, which contains rankings for these systems. Table 7 is organized to present the rating factors by groups in order of importance. Accessibility is first, since a system not accessible is unacceptable, even if technically excellent. Applicability ranks second only to accessibility since it determines if a system meets the criteria. Ease of use and documentation then follow, as they only are important considerations for systems ranked acceptable by the preceeding two groups. Performance is the last group of factors since performance only is a consideration for systems which are otherwise acceptable. Thus, as Table 7 shows, GCSS is the only system readily accessible to AFWAL, and since it is technically acceptable, GCSS becomes the system of choice.

While GCSS is not outstanding in its performance of any requirement, it is a competent system presenting no problems for AFWAL. DSDS is a better system than GCSS, and it is the only system having adequate documentation. However, it is run in a batch mode on a dedicated IBM 370, and is implemented in an

interactive mode on a Prime 400. Thus, it is not hosted on a computer available to AFWAL, and cannot be recommended. DAS/DDPM is by far the fastest system, but it also is implemented on a computer not available to AFWAL. Additionally, it is a proprietary product written in a proprietary language, further diminishing its attractiveness.

5.2 RECOMMENDATIONS

None of the systems individually meet all of the desired operational characteristics and requirements. Therefore, the following recommendation is presented. This will provide an interim near-term capability while concurrently defining AFWAL's ultimate long-term needs. A three-step approach is recommended:

1) GCSS Implementation - Since GCSS is government property and accessible via the ARPANET, if offers a near term and low cost means for meeting AFWAL's immediate needs. To implement the system will require obtaining the necessary authorizations to access the NADC CDC 6600 support from NADC personnel to learn the interactive system and the necessary hardware to accomplish the interface.
2) Generate Requirements - The purpose of this activity is to define specific AFWAL/AAAS simulation and modeling needs and requirements. It is recommended that a definitive set of requirements based upon current state-of-the-art techniques and long-term AFWAL goals be derived. This will provide AFWAL a baseline from which it can accurately assess the candidate systems in terms of specific requirements.
3) Detailed Evaluation and Assessment - Here it is recommended that AFWAL review and assess the capability of the three candidate systems (GCSS, DSDS, and DAS/DDPM) to meet the requirements

specified in (2), and determine the feasibility and scope of modifying each system to meet the requirements. Trade off the results against developing a new system to meet the requirements.

4) In parallel with 1), 2), and 3), conduct a dialog with RADC and ESD/MITRE to determine if there are functions and capabilities that are being acquired by these agencies that could be shared to the mutual benefit of all.

TABLE 7. RATING FACTORS FOR GCSS, DSDS, AND DAS/DDPM

	GCSS	DSDS	DAS/DDPM
ACCESSIBILITY	*	O	-
Language	+	+	- (proprietary)
Host Computer/Portability	+	-	-
Operational Status	+	+	+
All Software	+	+	+
Availability	*	+	- (proprietary)
APPLICABILITY	+	+	+
Avionics Applications	+	+	+
Distributed Processing	+	+	+
Information Transfer	+	+	+
Fault Tolerance	+	+	-
Time Division Multiplex	+	+	-
EASE OF USE	O	+	O
Set-Up	+	O	O
Input Language	-	O	O
Diagnostics	O	O	-
Flexibility	+	+	+
Outputs & Reports	+	+	+
DOCUMENTATION	O	+	-
Systems Specifications	O	+	-
Interface Descriptions	-	-	-
User's Guide	+	*	-
PERFORMANCE	+	+	*
Tcpu/Ts	O	+	*
Validity	+	+	+
Capacity	+	+	+

* = The system exceeds AFWAL needs for this capability.
+ = The system meets AFWAL needs for this capability.
O = The system minimally satisfies AFWAL needs for this capability.
- = The system does not satisfy AFWAL needs for this capability.

SECTION VI

COMPENDIUM

The following lists the simulations that, during the course of the survey, exhibited the most potential to meet AFWAL's needs. Included in the list are those simulations that survived the screening as well as those that were dropped from consideration. Each entry is given in a quick reference format with source, contact, and a brief description. An asterisk next to the title indicates the systems that were discussed in Section IV.

<u>Title</u>

VANS - Value Added Network Simulator

<u>Source</u>

University of Minnesota, Computer Sciences Department, Minneapolis, Minnesota

<u>Contact</u>

Dr. G. Michael Schneider

<u>Description</u>

VANS is a system designed to investigate the performance of communications networks, message traffic handling and protocols. It is hosted on a CDC 6600 and written in PASCAL and SIMULA. The system was not designed to handle large networks. Thus its data structures overflowed when models size exceeded twenty nodes or more. The system was not documented nor could support from the designer be obtained. This, along with the system's inability to handle large models, resulted in dropping the system from consideration.

<u>Title</u>

ARES - Automated Requirements Engineering System

<u>Source</u>

Martin Marietta Aerospace Company, Denver, Colorado

Contact

Mr. Richard E. Wachs

Description

ARES is a discrete event simulation that models operating systems, data base management, applications processing and man-machine interactions. ARES was one of the four systems that was included in the ESD/MITRE evaluation. The system is hosted on a CDC 6500 and a DEC 11/70. It is written in PASCAL. ARES was not retained principally because of its current status. The system was not completely designed nor was user or descriptive documentation completed.

Title

PERCAM - Performance and Configuration Analysis Methodology

Source

TRW, Inc., Defense and Space Systems Group, McLean, Virginia

Contact

Mr. D. L. DeHaven

Description

PERCAM is a discrete event simulation that simulates network data flows. The system is used primarily in assessing combat effectiveness by the Army in war-gaming. The system is basically a modeling tool that facilitates model development. The system is hosted on a CDC 6600 and is programmed in FORTRAN. It was dropped from consideration primarily because of the absence of basic functions required for the avionics functions and the anticipated effort that would be required to adapt it to this environment. PERCAM was also one of the systems included in the ESD/MITRE evaluation.

Title

*GCSS - Generalized Computer System Simulator

Source

Naval Air Development Center, Warminster, Pennsylvania

Contact

C. Mattes, W. Garrison

Description

GCSS is a government owned all-software simulation designed to evaluate alternate computer and system architectures and bussing arrangements. It is hosted on the CDC 6600 computer and is written in SNOBOL and SIMSCRIPT I.5. GCSS will be used to investigate architecture alternatives and signal processing requirements for the V/STOL avionics system.

Title

*DSDS - Data System Dynamic Simulator

Source

NASA/Marshall Space Flight Center (MSFC), General Electric Company (GEC), Huntsville, Alabama

Contact

John R. Piner, NASA/MSFC; Norman F. Geer, GEC

Description

DSDS is a government owned all-software simulation designed to perform comprehensive design analysis and trade studies of large data processing and communication systems. The batch version is hosted on the IBM 360/75 and the interactive version is hosted on PRIME 400. DSDS is written in FORTRAN with a modified version of GASP IV for control logic. The basic concept of DSDS is the library of models that can be interconnected to form a network configuration for evaluation and analysis.

Title

*DAS/DDPM - Design Analysis System/Distributed Data Processing Model

Source

Hughes Aircraft Company, Ground Systems Group, Fullerton, California

Contact

Mr. John Camp; Mr. Robert Jones

Description

DAS/DDPM is a proprietary simulation and modeling system used to analyze distributed data processing design alternatives. The

system is hosted on an AMDHAL 470/v6 and is used interactively. The system is programmed in a Hughes proprietary language - PLS (Product Line Simulation). DAS/DDPM is one of the four systems included in the ESD/MITRE evaluation.

REFERENCES

The following lists citations of documents pertaining to simulation and modeling that were reviewed , in search of relevant systems. Each entry is presented by title, sponsor/author, source and date.

1. "Installation Validation and Support of the Generalized Computer System Simulator, Final Users Manual", Contract No. N62269-77-C-0179 (Sponsored by NADC, Warminster, PA.), Honeywell Aerospace and Defense Group April 19, 1978

2. "Installation Validation and Support of the Generalized Computer System Simulator, GCSS Source Programs, Contract No. N62269-77-C-0179 (Sponsored by NADC, Warminster, PA.), Honeywell Aerospace and Defense Group April 7, 1978

3. "DAS: An Automated System to Support Design Analysis" and "Design Evaluation of Distributed Data Bases and Data Base Management", Willis, R. R., Hughes Aircraft Company, Ground Systems Group, Fullerton, California, January 1980

4. "Spacelab Simulation, DSDS, SL-1 Model Implementation Report", Contract No. NAS8-31532 (Sponsored by NASA/MSFC), General Electric Company, Huntsville Operations of the Space Division, Huntsville, Alabama, March 1979

5. "Study to Establish Models and Simulations for Data Systems, Volume 2, Users Manual", Contract No. NA8-31532 (Sponsored by NASA/MSFC), General Electric Company, Space Division, Huntsville, Alabama, May 17, 1976

6. "Data System Dynamic Simulator Training Manual", General Electric Company, Space Division, Huntsville, Alabama, September 1978

7. "Simulation Analysis Report, Communications Network Design Problem", Contract No. NAS8-32983 (Sponsored by NASA/MSFC and USAF/ESD), General Electric Company, Huntsville Operations of the Space Division, Huntsville, Alabama, October 1979

8. "Air Defense System Performance Analysis Final Report, Volume 3, PERCAM Users Manual", Contract No. DAA40-77-C-0006 (Sponsored by U.S. Army Missile Research and Development Command, Redstone Arsenal, Alabama), TRW Inc., Defense and Space Systems Group, Huntsville, Alabama, December 30, 1977

9. Exerpts from the "ARES Users Guide", Wachs, R. E., Martin Marietta Aerospace, Denver, Colorado, January 1980

10. "Evaluation of General Purpose Modeling Systems", Contract No. F19628-80-C-0001 (Sponsored by USAF/ESD), Shultze H.P., MITRE Corporation, Bedford, Massachusetts, February 1980

BIBLIOGRAPHY

1. David L. Adamy, Electronic Warfare/Defense Electronics, 11:71-72, June 1979. (AFWAL)

2. J. S. Annino, and Edward Russell, "The Ten Most Frequent Causes of Simulation Analysis Failure - And How To Avoid Them!", Simulation, June 1979. (AFIT)

3. J. S. Annino, and E. C. Russell, "Successful Simulation of Complex Systems with Simscript II.5", Simuletter, Fall and Winter 1978-79. (WSU: T57.62.553)

4. M. Arozullah, John Isler, Wayne Densmore, and Ray Washburn, "Simulation of a Distributed Processor Computer Communication Network", Proceedings of the 7th Annual Pittsburgh Conference on Modeling and Simulation, University of Pittsburgh, Pennsylvania, April 26-27, 1979. (WSU: TA343.P582)

5. P. T. Beatty, et al., "An Introduction to the Simulation of a Multiple CPU Military Communications System", Simuletter, July 1976, Proceedings of the Symposium of Computer Systems. (WSU: T57.553)

6. Leland Blank, "A Simulator for Multilevel Distribution Systems", Simulation, February 1979, p. 54-62. (AFIT)

7. L. S. Bonderson, "System Properties of One-Dimensional Distributed Systems", Journal of System Measurement and Control, 99:85-90, July 1977. (WSU: TA1.J623)

8. W. E. Bracker, and E. R. Sears, "Test Facility for a Message-Switching System", Bell System Technical Journal, Vol. 55, No. 7, p. 857-73, September 1976. (AFWAL)

9. G. A. Brent, and T. M. McCalla, Jr., "Exploring Team Avionics by Simulation", Eleventh Annual Simulation Symposium, Tampa, Florida, 1978, p. 155-170. (WSU: T57.S57)

10. James C. Brown, K. M. Chandy, R. M. Brown, Tom W. Keller, Donald F. Towsley, and C. William Dissly, "Hierarchical Techniques for the Development of Realistic Models of Complex Computer Systems", Proceedings of the IEEE, June 1975, p. 966-975. (WSU: TK5200.I6)

11. J. Buck, et al., "Man in the Loop Simulation", Simulation, May 1978. (AFIT)

12. Malcolm D. Calhoun, A Study of Two Avionics Multiplex Simulation Models, Air Force Avionics Laboratory, Wright-Patterson Air Force Base, Ohio 45433, 1979.

13. B. J. Carey, "A Communication Systems Simulation Laboratory", IEEE Transactions on Education (USA), Vol. E-18, No. 4, p. 189-95, November 1975. (WSU: T61.I2)

14. Cellier and Blity, "Gasp V - A Universal Simulation Package" Simulation of Systems, 8th Congress, Delft, Netherlands, 1976. (WSU: QA76.9.C65.S542)

15. K. M. Chandy, et al., "Distributed Simulation of Networks", Computer Networks, April 1979. (WSU: TK5105.5.C53)

16. Wesley W. Chu, David Lee, and Brandon Iffla, "Distributed Processing System for Naval Data Communication Networks", AFIPS NCC Expo, Anaheim, California, 1978. (OSU)

17. George Consolver, and M. Moore, Distributed Processor/Memory Architecture Design Program, DDC No.: AD A016482.

18. P. C. Cooley, "The Underlying Structure of Simulation Problems and Simulation Software", 8th Annual Simulation Symposium, Tampa, Florida, March 1975, p. 45-55. (WSU, AFIT: T57.62.S57)

19. B. Coulbeck, and M. J. H. Sterling, "Modelling Techniques in Dynamic Control of Water Distribution Systems", Measurement and Control, October 1978. (WSU: TA1.M4)

20. Tom Cunningham, et al., Fault Tolerance Digital Flight Control with Analytical Redundancy, May 1977, DDC No.: AD A045671.

21. I. Dekker, ed., Simulation of Systems, 8th AICA Congress, Delft, Netherlands, 1976. (WSU: QA76.9.C65.S542)

22. J. J. Digiovanni, and G. B. Jordan, EW Simulation-Advantages and Recent Advances (and a Tool for Operator Training and System Evaluation), Anetkna Inc., Mountain View, California, September 1974.

23. J. Fong, and C. Pottle, "Simulation of a Parallel Microcomputer System for Circuit Design", 1977 IEEE International Symposium on Circuits and Systems Proceedings, April 1975, p. 131-4. (AFIT: TK7801.I22)

24. Eric Foxley, "Hybrid Computer Performance Modelling System", Computer Journal, August 1978, p. 205-9. (AFIT)

25. E. F. Gehringer, and Herbert D. Schwetman, "Run-Time Characteristics of a Simulation Model", Simuletter, July 1976, (Proceedings of the Symposium on Simulation of Computer Systems). (WSU: T57.62.S53)

25. J. E. Heller, "Simulation of a Digital Transmission Line", Proceedings of the 6th Annual Pittsburgh Conference on Modelling and Simulation, University of Pittsburgh, Pennsylvania, 1975, p. 289-90. (WSU: TA343.P582)

26. ITT Avionics Division, Nutley, New Jersey, New Threats Simulator (NETS) Feasibility Study, August 1978.

27. M. S. Jayakumar, and T. M. McCalla, "Simulation of Microprocessor Emulation Using GASP-PL/I", Computer, 10:12-17, April 1977. (AFIT)

28. Ken-Shaw Lu, Failure Prediction for an On-Line Maintenance System in a Poison Shock Environment, DDC No.: AD A057446.

29. George A. Kilgore, Probabilistic Measures of Compromise, DDC No.: AD A037302.

30. Norihisa Komoda, and Koichi Haruna, Accessibility and Maintainability in Man-Machine Interactive Structural Modelling, Hitachi Ltd., 1978. (AFIT: Q300.I485a)

31. Frank J. Langley, Joan Sheehan, and Gerald LaGro, "Simulating Modular Microcomputers", Simulation, May 1979. (AFIT)

32. G. E. LaPara, and G. E. Whitehouse, "A-GERT and GPSS: User's Comparison", Proceedings of the 9th Annual Pittsburgh Conference on Modelling and Simulation, University of Pittsburgh, Pennsylvania, 1978. (WSU: TA343.P582)

33. Robert L. Leech, EQM Models for the Analysis and Design of a Computer Network of Functionalized Processors, DDC No.: AD A045721.

34. H. C. Lessing, System Technology Studies: Simulation Methods for the Validation of Digital Flight Control Systems, Ames Research Center, May 1978, DDC No.: VN 970133.

35. B. A. Lichtig, "Simulation is 'Unreal'", Simuletter, (July 1975, Proceedings of the Symposium on Simulation of Computer Systems). (WSU: T57.62.553)

36. Massachusetts Institute of Technology, Simulation of Packet Communication Architecture Computer Systems, DDC No.: AD A048290.

37. G. E. Miles, R. M. Pert, and A. A. B. Pritsker, "Crops; A GASP IV Based CROP Simulation Language", 1976 Summer Computer Simulation Conference, July 1976, Washington, D.C., p. 921-4. (WSU)

38. Naval Postgraduate School, Monterey, California, A Q-GERT Model and Analysis of the Communications in a Mechanized Brigade Covering Force, March 1979, DDC No.: AD A068526.

39. S. S. Nelson, "CONSIM: A Study of Control Issues in Conversational Simulation", The Computer Journal, May 1979. (WSU: QA76.C57)

40. J. K. Peacock, et al., "Distributed Simulation Using a Network of Processors", Computer Networks, February 1979. (WSU: TK5105.C53)

41. U. W. Pooch, "A Communications-Computer Simulation System", College Station, Texas, 1976 Summer Computer Simulation Conference, Washington, D.C., July 1976, p. 822-5. (WSU: 22:621.38195)

42. W. L. Price, "Data Network Simulation. Experiments at the National Physical Laboratory", Computer Networks, May 1977. (WSU: TK5105.C53)

43. W. L. Price "Simulation Studies of Data Communication Networks Operating in Diagram Mode", Computer Journal, 21:219-23, August 1978. (AFIT)

44. C. C. Reames, and M. T. Liu, "Design and Simulation of the Distributed Loop Computer Network (DLCN)", Symposium on Computer Architecture, Ohio State University, Columbus, Ohio, 1976. (WSU: TK788.A1S85)

45. N. P. Reddy, T. A. Krouskop, and P. H. Newell, Jr., "A Computer Model of the Lymphatic System", Computer Biology and Medicine, July 1977, p. 181-97. (AFIT)

46. M. Reiser, "A Queueing Network Analysis Computer Communication Networks with Window Flow Control", IEEE Transactions on Communications, August 1979. (WSU: TK5101.A1I2)

47. Research Triangle Institute Research, AFAL Simulation Facility/ Capability Manual, Vol. 1, Executive Summary and Systems, Avionics Division, Triangle Park, N.C., June 1977.

48. L. F. Robinson, "How GASP, Simula, and Dynamo View a Problem", 5th Annual Simulation Symposium, Tampa, Florida, p. 167-214. (WSU: T57.62557)

49. C. H. Sauer, and E. A. McNair, "Queueing Network Software for Systems Modelling", Software-Practice and Experience, May 1979. (WSU: QA76.5.56353)

50. G. M. Schneider, "A Modular Approach to Computer Network Simulation", Computer Networks, September 1976. (WSU: TA343.55)

51. G. M. Schneider, "Modeling Package for Simulation of Computer Networks", Simulation, December 1978, p. 181-192. (WSU: TA343.55)

52. R. L. Schwing, "The Man Rating Associated with the AFFDL LaMars System", AIAA Visual and Motion Simulation Conference, 1976. (AFWAL)

53. R. S. Sidell, and D. N. Wormley, "Efficient Simulation Method for Distributed Lumped Fluid Networks", Journal of Dynamic System Measurement and Control, May 1977, p. 34-40. (WSU: TA1.J623)

54. C. E. Sigal, and A. A. B. Pritsker, "Smooth: A Combined Continuous Discrete Network Simulation Language", Simulation, Vol. 22, No. 3, P. 65-73, March 1974. (AFIT)

55. A. J. Surkan, "Simulation of Storm Velocity Effects on Flow From Distributed Channel Networks", Water Resources Research, December 1974. (WSU: TC1.W3)

56. A. Thesen, "Evolution of a New Discrete Event Simulation Language for Inexperienced Users (WIDES)", Software Practice and Experience July 1977, p. 519-533. (WSU)

57. K. P. Tognetti, and C. Brett, "Simscript II and Simula '67-A Comparison", Australian Computer Journal, May 1972, p. 50-57. (WSU)

58. Ultra Systems, Incorporated, "Analysis of the Survivability of Shuttle (ALT) Fault-Tolerant Avionics System", NTIS No.: N7622288-89.

59. Yaacov L. Varoletal, "Applying Simulation to Highway Intersection Design", Simulation, April 1977. (AFIT)

60. Hoyt M. Warren, Jr., A General Computer Network Simulation Model, Air Force Institute of Technology, School of Engineering, Wright-Patterson Air Force Base, Ohio 45433, March 1977. (AFIT)

61. Jack Wellin, and Lawrence Eisenberg, "Simulation of Traffic Flow Using Electrical Network Parameters", Journal of the Franklin Institute, December 1974, p. 423-32. (AFIT)

62. P. R. West, "A Multi-Radar Tracking Simulation Using Algol 68R", Sigplan Notices, p. 175-181, June 1977. (WSU: QA76.A84)

63. Larry D. Wittie, "A Reconfigurable Microcomputer Network For Distributed Systems Research", Simulation, p. 145-53, November 1978. (AFIT)

64. D. B. Wortman, "Simulation with GASP IV: A Combined Distrete-Continuous Simulation Language", 9th Annual Simulation Symposium, Tampa, Florida, March 1976, p. 47-59. (WSU: T57.62.S57)

65. R. E. Young, and A. A. B. Pritsker, "GASPL/I: A PL/I Based Simulation Language", 1974 Winter Simulation Conference, Washington, D.C., January 1974, Vol I, p. 25-35. (OSU)

*U.S.Government Printing Office: 1980 — 757-002/191

LMED

8